



# Using the TotalView Debugger and MemoryScape

Aug 21, 2013

Jennifer Locke, Rogue Wave



# Agenda



- TotalView Overview
- Starting up TotalView on Pleiades
- Accessing TotalView Remotely
- MemoryScape Leak Detection Example on Pleiades
- Q&A



# What is TotalView?

## A comprehensive debugging solution for demanding parallel and multi-core applications

- Wide compiler & platform support

- C, C++, Fortran 77 & 90, UPC
- Unix, Linux, OS X
- CUDA GPU, Intel MIC

- Handles Concurrency

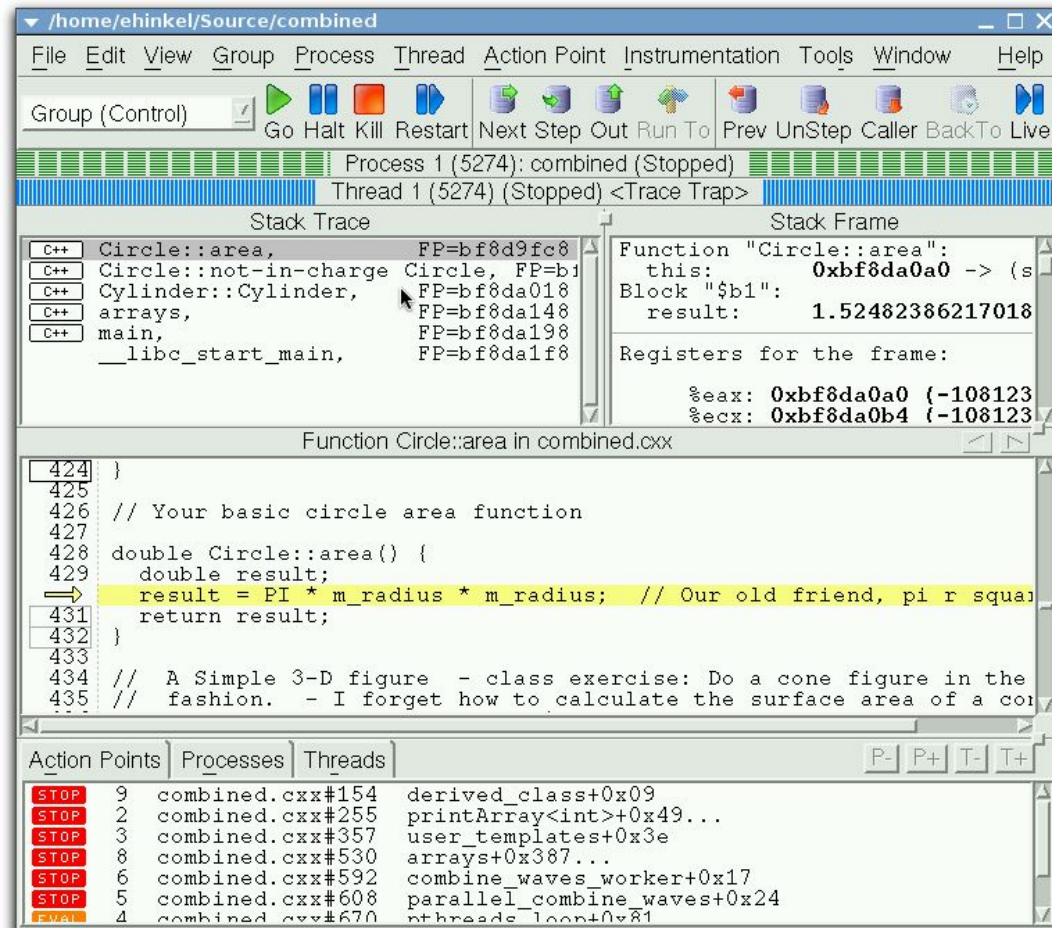
- Multi-threaded Debugging
- Parallel Debugging
  - MPI, PVM, OpenMP
- Remote and Client/Server Debugging

- Integrated Memory Debugging

- Reverse Debugging

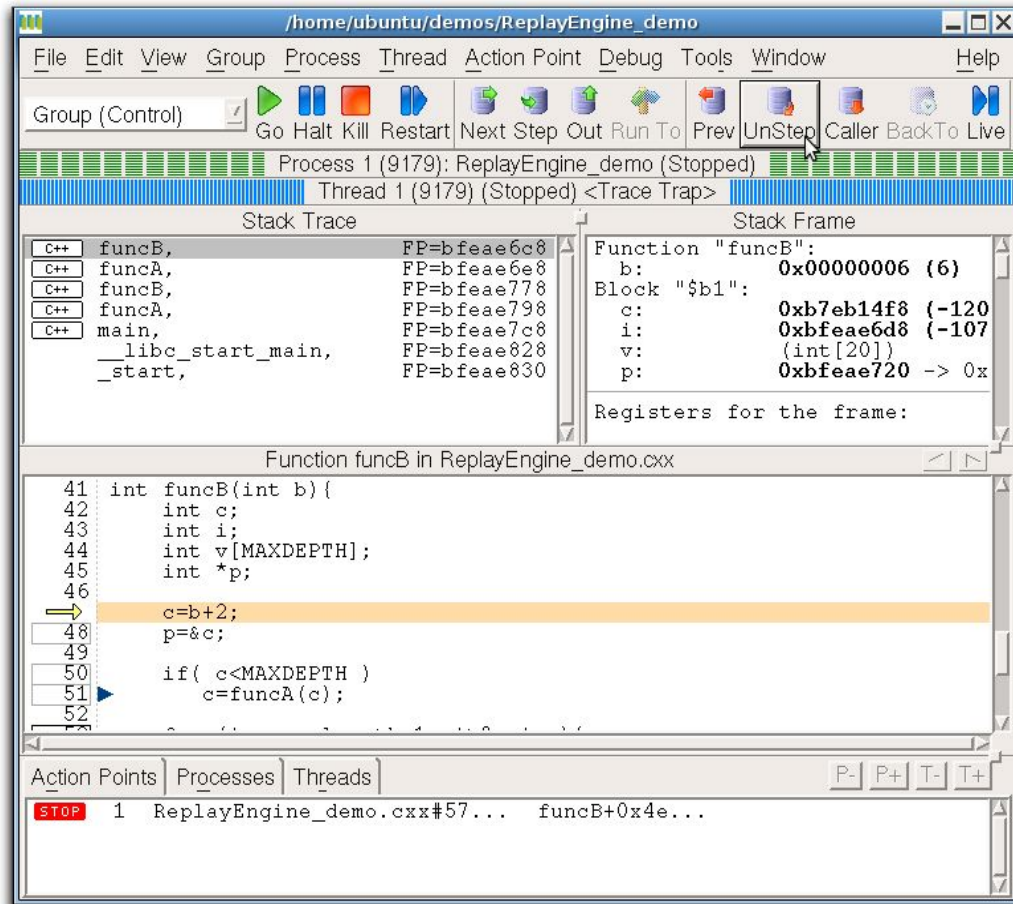
- Supports a Variety of Usage Models

- Powerful and Easy GUI / Visualization
- CLI for Scripting
- Long Distance Remote Debugging
- Unattended Batch Debugging





# Reverse Debugging - ReplayEngine



- **Captures execution history**

- Records all external input to program
- Records internal sources of non-determinism
- Turn it on at any point

- **Replays execution history**

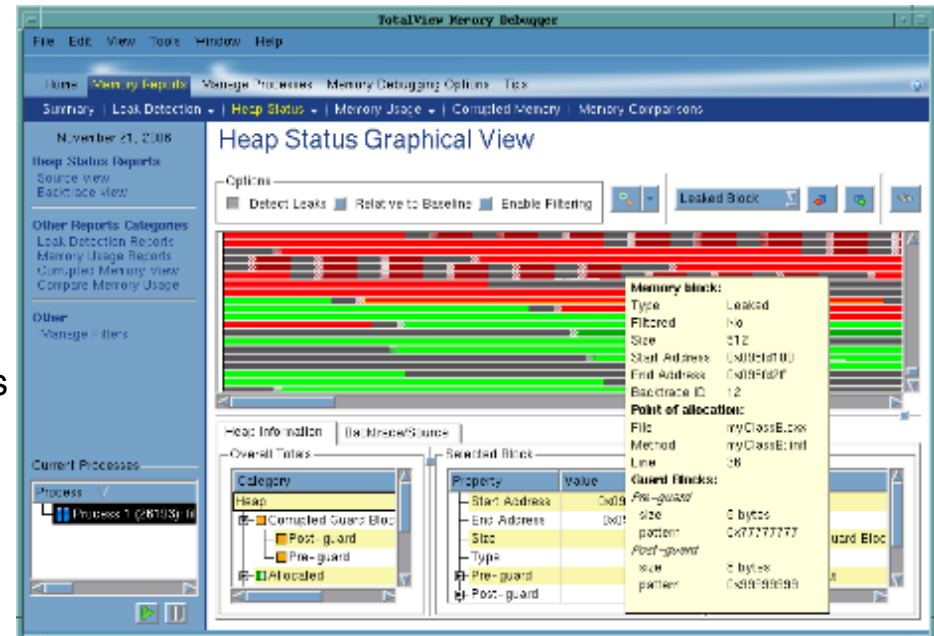
- Examine any part of the execution history
- Step back as easily as forward
- Jump to points of interest



# MemoryScape Overview



- Runtime Memory Analysis : Eliminate Memory Errors
  - Detects memory leaks *before* they are a problem
  - Explore heap memory usage with powerful analytical tools
  - Use for validation as part of a quality software development process
- Major Features
  - Included in TotalView, or Standalone
  - Detects
    - Malloc API misuse
    - Memory leaks
    - Buffer overflows
  - Supports
    - C, C++, Fortran
    - Linux, Unix, and Mac OS X
    - MPI, pthreads, OMP, and remote apps
  - Low runtime overhead
  - Easy to use
    - Works with vendor libraries
    - No recompilation or instrumentation



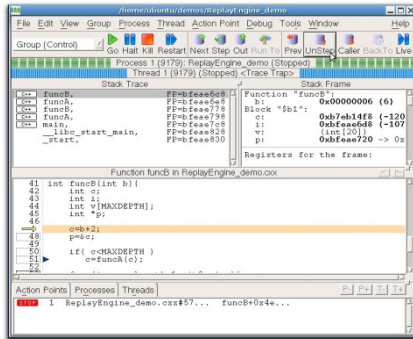




# TotalView Debugging Ecosystem

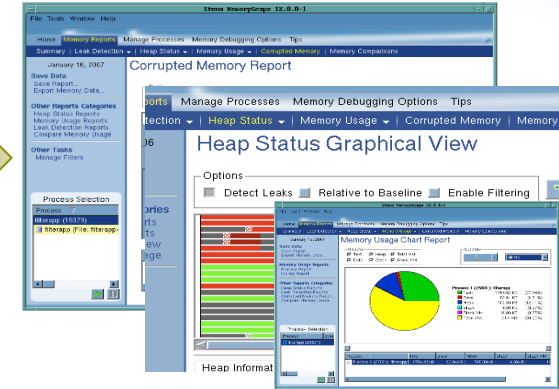
## Debugging with TotalView

### Reverse Debugging with ReplayEngine



- Captures execution history
- Replays execution history
- Enable 'on Demand'
- Step backwards!

### Memory Debugging with MemoryScape



- Graphical View of Heap Memory
- Low Overhead
- Detect:

- Leaks
- Buffer over/underflow

- MPI memory debugging

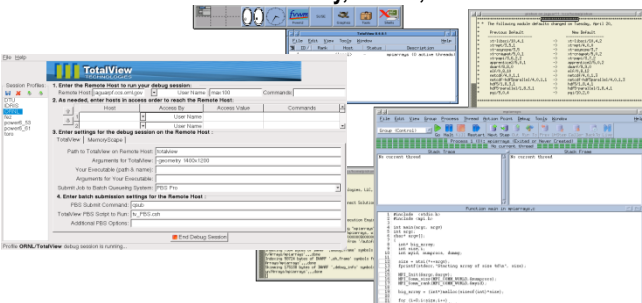
- Parallel debugging
- Accelerator and coprocessor debugging
- Wide compiler and platform coverage
- Work Graphically
- Troubleshoot even your hardest bugs
- Develop Code Confidently!

### Remote Display Window

Easy, Secure, Fast

### Batch Debugging with TVScript

- Unattended TotalView debugging



The Debugger of Choice for HPC and Enterprises



# More Information



TotalView demonstration videos available on the  
Rogue Wave TotalView Products page

<http://www.roguewave.com/products/totalview/resources/videos.aspx>



# Starting TotalView on Pleiades (1/1)



- Load Modules
  - TotalView
    - module load *totalview/8.12.0-0*
  - MPI
    - module load *mpi-sgi/mpt.2.06r6*
      - Latest version of SGI MPT library *mpi-sgi/mpt.2.08r7* contains `malloc_intercept` which blocks TotalView Memory debugging
      - Module *mpi-sgi/mpt.2.06r6* was tested and allows TotalView MemoryScape to properly track heap allocations



# Starting TotalView on Pleiades (2/2)



- Set TotalView Environment Variable \$TVLIB
  - TotalView
    - `setenv TVLIB /nasa/totalview/toolworks/totalview.8.12.0-0/linux-x86-64/lib`
  - Compile MPI programs with TV HIA library
    - `mpicc -g -o yourProgram yourprogram.ext -L$TVLIB -ltvheap_64 -Wl,-rpath,$TVLIB`
      - Required for MPI programs
      - HIA can be dynamically loaded in single process applications
  - Execute MPI program in TotalView
    - `mpiexec_mpt -tv -n8 ./yourProgram`
      - Do not select “Enable Memory debugging”
      - Executable is linked to the HIA library tvheap



# Accessing Pleiades Remotely



- TightVNC
  - Follow the same Starting TotalView instructions on slides 8 and 9
- TotalView Remote Display Client (RDC)
  - RDC can be downloaded from Rogue Wave Website at <http://www.roguewave.com/products/totalview/remote-display-client.aspx>



# TotalView Remote Display Client (RDC)



TotalView Remote Display Client

**TotalView TECHNOLOGIES**

Session Profiles:

- boater
- edge
- fedora12-x8664
- gooney6
- lanl
- lanl.batch
- loki
- macbookpro
- minnie
- ntkvnc
- nvidia5
- perseid
- pleiades**
- prodigy.totalviewtech.co
- rhel56
- rhel57
- ubunt1104-x8664
- vulcan

1. Enter the Remote Host to run your debug session:

Remote Host:  User Name:

2. As needed, enter hosts in access order to reach the Remote Host:

	Host	Access By	Access Value	Commands
1	sfe1.nas.nasa.gov	User Name	pmthomps	
2		User Name		

3. Enter settings for the debug session on the Remote Host :

Path to TotalView on Remote Host:

Arguments for TotalView:

Your Executable (path & name):

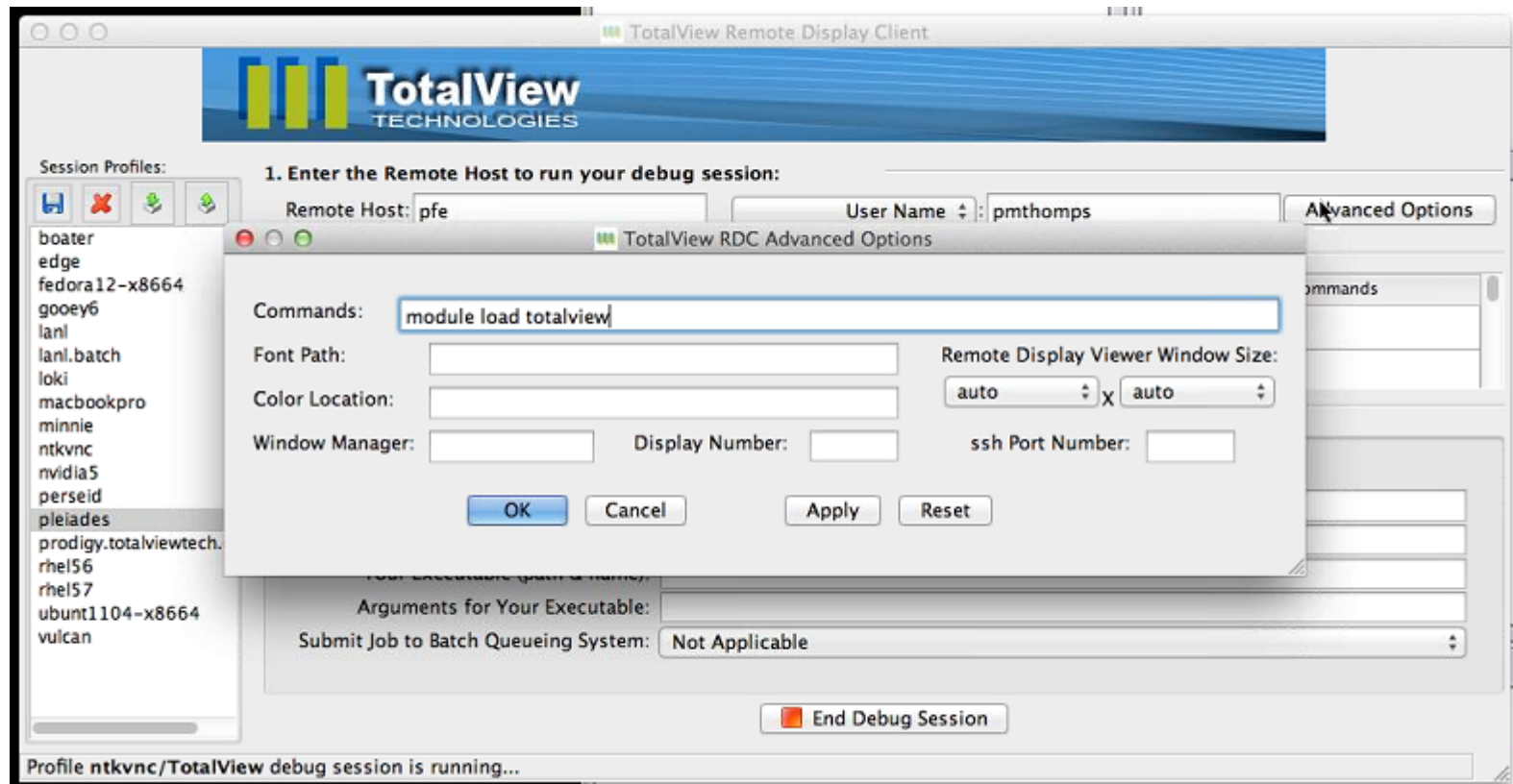
Arguments for Your Executable:

Submit Job to Batch Queueing System:

No session running



# Pre-load TotalView Module on RDC







# Memory Debugging



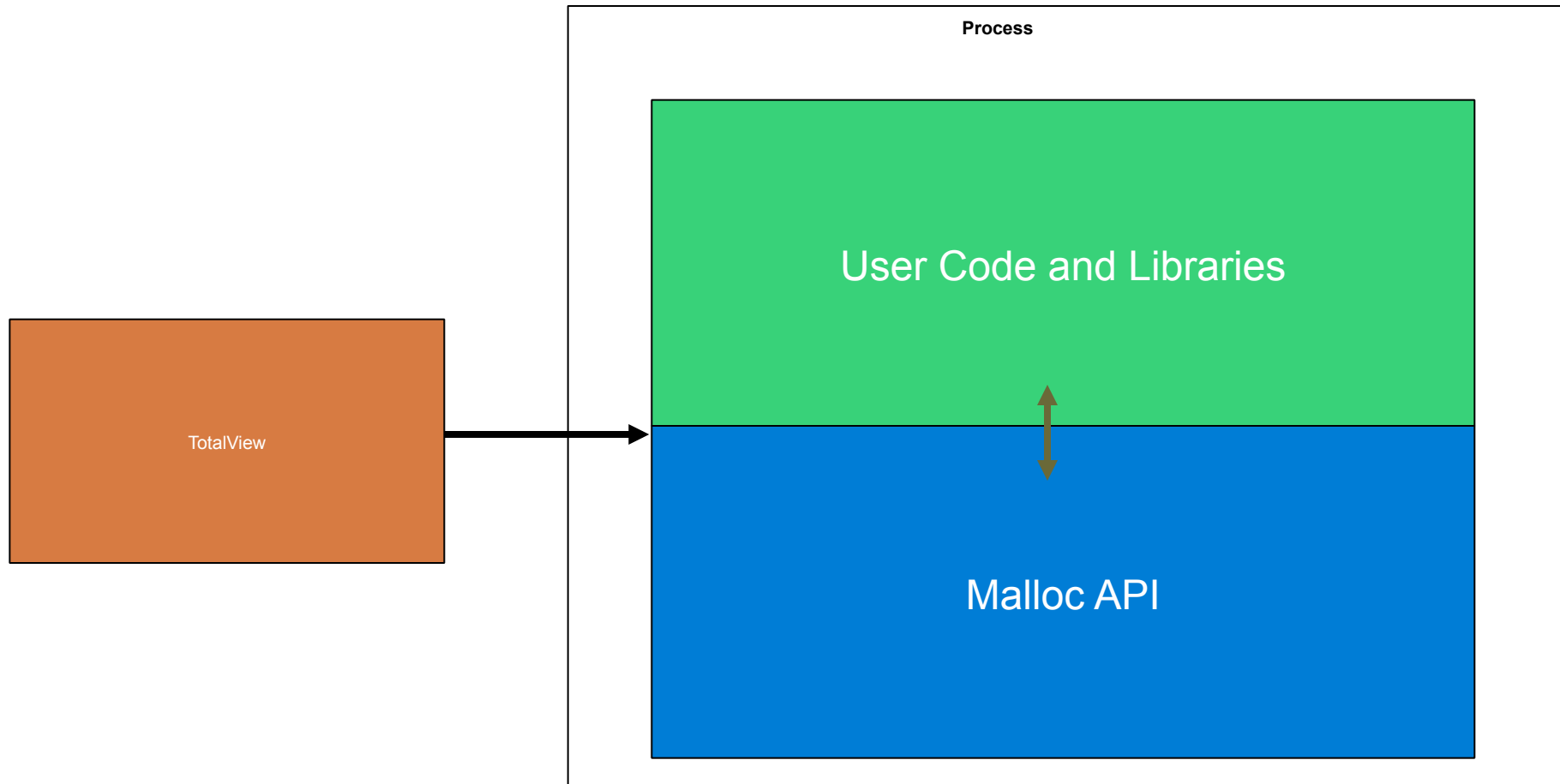


# What is a Memory Bug?

- A Memory Bug is a mistake in the management of heap memory
  - Failure to check for error conditions
  - Leaking: Failure to free memory
  - Dangling references: Failure to clear pointers
  - Memory Corruption
    - Writing to memory not allocated
    - Over running array bounds

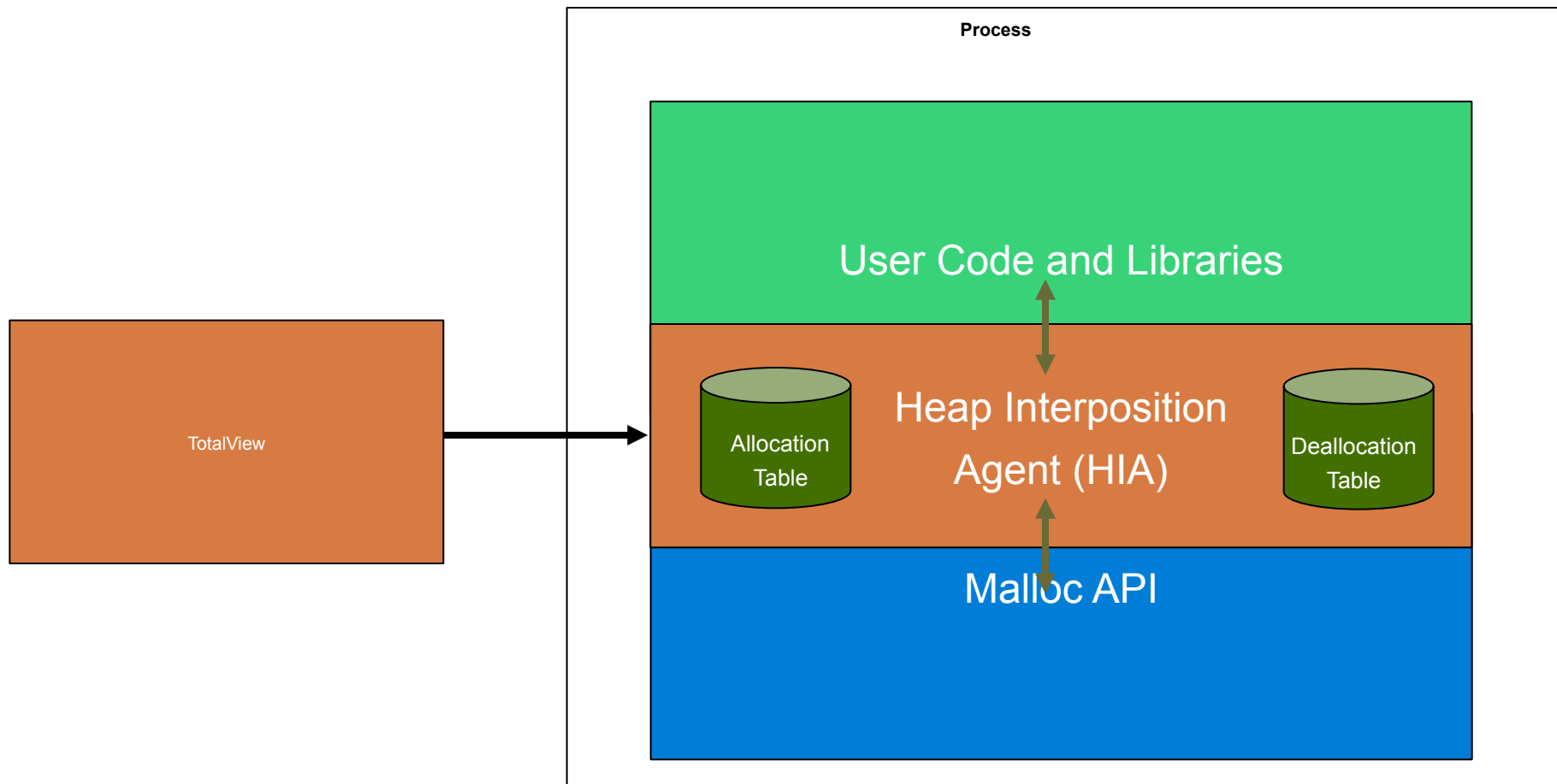


# The Agent and Interposition





# The Agent and Interposition





## Advantages of TotalView HIA Technology

- Use it with your existing builds
  - No Source Code or Binary Instrumentation
- Programs run nearly full speed
  - Low performance overhead
- Low memory overhead
  - Efficient memory usage
- Support wide range of platforms and compilers



# Memory Debugger Features



- Automatic detection of allocation problems
- Graphical heap view
- Leak detection
- Block painting
- Memory Hoarding
- Dangling pointer detection
- Deallocation/reallocation notification
- Memory Corruption Detection - Guard Blocks
- Memory Comparisons between processes
- Collaboration features



# MemoryScape Leak Detection Example



```
else
{
    h  = 1.0 / (double) n;
    sum = 0.0;
    x = f(h);
    for (i = myid + 1; i <= n; i += nuxprocs)
    {
        /* comments to increase line number of malloc */
        /*
        *
        */

        x = h * ((double)i - 0.5);
        sum += f(x);
        freeptr = (char *) malloc ((unsigned long)150 * sizeof(char));
        strcpy(freeptr, "Testin");

    }
    mypi = h * sum;

    MPI_Reduce(&mypi, &pi, 1, MPI_DOUBLE, MPI_SUM, 0, MPI_COMM_WORLD);
}
```

(c) 1999



The screenshot displays the TotalView IDE interface. A 'Startup Parameters - \_mpirun\_' dialog box is open, showing the 'Debugging Options' tab. The 'Arguments' tab is also visible. The 'Debugging Options' section includes checkboxes for 'Enable ReplayEngine', 'Enable memory debugging', 'Halt on memory errors', and 'Enable CUDA memory checking'. The 'Arguments' section includes a checkbox for 'Show Startup Parameters when TotalView starts'. The 'ProcessWindow' is open, showing the 'winTitle' as 'No current thread' and the 'Stack Frame' as 'No current thread'. The 'Source' editor is open, showing the file 'libxmpi.so'. The 'Source' editor also displays the following text:

```
libxmpi.so'...done  
b/libxmpi.so'...done  
libxmpi.so'...done  
loaded at 0xff00000090000009  
loaded at 0xff0000009006150  
loaded at 0xff000000901be7  
loaded at 0xff000000901  
initially loaded at 0xff00  
...done
```

The 'ProcessWindow' also shows the following text:

```
ProcessWindow  
Action Point Debug Tools Window Help  
Next Step Out Run To Record GoBack Prev UnStep Caller BackTo Live  
winTitle  
No current thread  
Stack Frame  
Source
```

The 'Source' editor also shows the following text:

```
File Edit View Tools Window  
ID/ Rank Host Status Descript  
... 1 <local> - _mpirun_ (0 active t
```

The 'ProcessWindow' also shows the following text:

```
Action Points Processes Threads  
P- P+ T- T+
```



TightVNC: pmthomps's X desktop (pfe23:91) (on prodigy.totalviewtech.com)

2028809.pbsp11 ganatthe normal node\_shuffle 1 1 00:10 Q 10:41 --

2028810.pbsp11 ganatthe normal node\_shuffle 1 1

2028811.pbsp11 ganatthe normal node\_shuffle 1 1

2028812.pbsp11 ganatthe normal node\_shuffle 1 1

2028813.pbsp11 ganatthe normal node\_shuffle 1 1

2028814.pbsp11 ganatthe normal node\_shuffle 1 1

2028815.pbsp11 ganatthe normal node\_shuffle 1 1

bash-3.2\$ pud

/u/pnthonps

bash-3.2\$ module list

bash: module: command not found

bash-3.2\$ exit

exit

%module list

No Modulefiles Currently Loaded.

%module1 module load mpi-sgi/npt,2.06r6

%module: Command not found.

% module load mpi-sgi/npt,2.06r6

%module load totalview/8.12.0-0

%which totalview

/nasa/totalview/toolworks/totalview,8.12.0-0/bin/totalview

%setenv TVLIB /nasa/totalview/toolworks/totalview,8.12.0-0/lib

%picc -g -o cpi.npt6 cpi.c -L\$TVLIB -ltvheap\_64 -Wl,-rpath,\$T

2

TotalView 8.12.0-0

File Edit View Tools Window

ID/	Rank	Host	Status	Descript
...	1	<local>	-	_mpirun_ (0 active t

\_mpirun\_

File Edit View Group Process Thread Action Point Debug Tools Window Help

Group (Control)

Go Halt Kill Restart Next Step Out Run To Record GoBack Prev UnStep Caller BackTo Live

Process 1 (0): \_mpirun\_ (Exited or Never Created)

No current thread

Stack Trace

No current thread

Stack Frame

No current thread

Function main in \_mpirun\_

Parallel program has not yet been started.

Action Points Processes Threads

P- P+ T- T+

sisu-login3

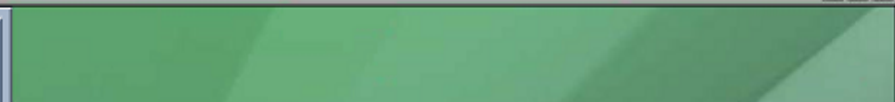
3.2 Constructing a batch job file | csc.fi - Mozilla Firefox



TotalView 8.12.0-0				
File Edit View Tools Window				
ID /	Rank	Host	Status	Description
1		<local>	-	_mpirun_ (0 active


 XTerm 1 2 3 4 sisu-login3 sisu-login3 3.2 Constructing a batch job file csc.fi - Mozilla Firefox





00:10.0 10:41 -- xterm

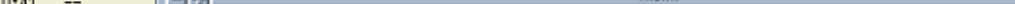
☐ No

Stack Frame


```
Thread is running
```

P- | P+ | T- | T+





```

1..done
initially loaded at 0xff000

1'...done
..done
initially loaded at 0xff

2.so'...done
o'...done
initially loaded at 0xff0

.so'...done
'...done

```

1043

[illegible]



TightVNC: pmthomp's X desktop (pfe23:91) (on prodigy.totalviewtech.com)

Desktop icons: Eyes, Fvwm2, openSUSE, Graphics, Tools, Shells.

Terminal window (X Desktop):

```
2028809.pbasp11 ganatthe normal node_shuffle 1 1 00:10 0 10:41 --
2028810.pbasp11 ganatthe normal node_shuffle 1 1
2028811.pbasp11 ganatthe normal node_shuffle 1 1
2028812.pbasp11 ganatthe normal node_shuffle 1 1
2028813.pbasp11 ganatthe normal node_shuffle 1 1
2028814.pbasp11 ganatthe normal node_shuffle 1 1
2028815.pbasp11 ganatthe normal node_shuffle 1 1
bash-3.2$ pwd
/u/pnthonps
bash-3.2$ module list
bash: module: command not found
bash-3.2$ exit
exit
$module list
No Modulefiles Currently Loaded.
$module load mpi-mpi/2.0.6r6
module: Command not found.
$ module load mpi-mpi/2.0.6r6
$module load totalview/8.12.0-0
$which totalview
/nasa/totalview/toolworks/totalview/8.12.0-0/bin/totalview
$setenv TVLIB /nasa/totalview/toolworks/totalview/8.12.0-0/lib
$mpicc -g -o mpi.mpi6 mpi.c -L$TVLIB -ltvheap_64 -Wl,-rpath,$TVLIB
```

Debugger window (\_mpirun\_<mpi.mpi6> 0):

File Edit View Group Process Thread Action Point Debug Tools Window Help

Group (Control) [X] Go Halt Kill Restart Next Step Out

Rank 0: \_mpirun\_<mpi.mpi6> 0  
Thread 1 (469125217256)

Stack Trace:

- pthread\_barrier\_wait, FP=7fffffffef110
- MPI\_SGI\_barrier, FP=7fffffffef0b0
- MPI\_SGI\_init, FP=7fffffffef150
- call\_init, FP=7fffffffef190
- \_dl\_init, FP=7fffffffef1f0

Function main in:

```
3 #include <math.h>
4 #include <stdlib.h>
5 #include <string.h>
6
7 double f( double );
8 double f( double a )
9 {
10     return (4.0 / (1.0 + a*a));
11 }
12
13 int main( int argc, char *argv[])
14 {
15     int done = 0, n, myid, numprocs, i;
16     double PI25DT = 3.141592653589793238462643;
17     double mypi, pi, h, sum, x;
18     double startwtime = 0.0, endwtime;
19     int namelen;
20     char processor_name[MPI_MAX_PROCESSOR_NAME];
21     char *freeptr;
22
23     MPI_Init(&argc,&argv);
24     MPI_Comm_size(MPI_COMM_WORLD,&numprocs);
25     MPI_Comm_rank(MPI_COMM_WORLD,&myid);
```

Debugger menu (Open MemoryScope):

- Enable ReplayEngine
- Previous (Alt+Shift+N)
- Unstep (Alt+Shift+S)
- Caller (Alt+Shift+O)
- Back To (Alt+Shift+R)
- Go Back (Alt+Shift+G)
- Live (Alt+Shift+L)
- Previous Instruction (Alt+Shift+X)
- Unstep Instruction (Alt+Shift+I)
- Enable Memory Debugging (Ctrl+Shift+M)
- Stop on Memory Errors
- Open MemoryScope (Alt+Shift+M)
- Heap Baseline
- Memory Block Properties
- Memory Event Details
- Enable CUDA Memcheck

Terminal window (TotalView 8.12.0-0):

ID/	Rank	Host	Status	Description
1	<local>	B		_mpirun_ (1 active th
2	0 <local>	T		_mpirun_<mpi.mpi6> 0
3	1 <local>	T		_mpirun_<mpi.mpi6> 1
4	2 <local>	T		_mpirun_<mpi.mpi6> 2
5	3 <local>	T		_mpirun_<mpi.mpi6> 3
6	4 <local>	T		_mpirun_<mpi.mpi6> 4
7	5 <local>	T		_mpirun_<mpi.mpi6> 5
8	6 <local>	T		_mpirun_<mpi.mpi6> 6
9	7 <local>	T		_mpirun_<mpi.mpi6> 7

Taskbar: XTerm, 1, 2, 3, 4, sisu-login3, sisu-login3, 3.2 Constructing a batch job file | csc.fi - Mozilla Firefox



# Root Window



- State of all processes being debugged
- Process and Thread status
- Instant navigation access
- Sort and aggregate by status

The screenshot shows the 'Etnus TotalView 7.1' application window. It has a menu bar with 'File', 'Edit', 'View', 'Tools', and 'Window'. Below the menu bar is a table with columns: 'ID', 'Rank', 'Host', and 'Status'. The table contains the following data:

ID	Rank	Host	Status
1	0	<local>	B
5		intrepid.etnus.c	T
6		intrepid.etnus.c	T
7		intrepid.etnus.c	T
8		intrepid.etnus.c	T
9		intrepid.etnus.c	T
10		intrepid.etnus.c	T
11		intrepid.etnus.c	T
12		intrepid.etnus.c	T
13	1	<local>	B
13.1	1	<local>	B4
14	2	<local>	B
15	3	<local>	B

## ► Status Info

- T = stopped
- B = Breakpoint
- E = Error
- W = Watchpoint
- R = Running
- M = Mixed
- H = Held



# TotalView Root Window

The screenshot shows the Etnus TotalView 7.1.0-0 Root Window. The window has a menu bar (File, Edit, View, Tools, Window, Help) and a table with columns: ID, Rank, Host, Status, and Description. The table lists various processes and threads. Callouts point to specific features:

- Hierarchical/Linear Toggle:** Points to the icon in the top-left corner of the table.
- Host name:** Points to the 'Host' column.
- Rank # (if MPI program):** Points to the 'Rank' column.
- TotalView Thread ID #:** Points to the 'ID' column.
- Expand - Collapse Toggle:** Points to the expand/collapse icon next to the ID column.
- Process Status:** Points to the 'Status' column.
- Action Point ID number:** Points to the 'B4' status value for thread 13.1.

ID	Rank	Host	Status	Description
1	0	<local>	B	mismatchLinux.0 (1 active threads)
5		intrepid.etnus.c	T	/home/barryk/tests/fork_loopLinux (5
6		intrepid.etnus.c	T	/home/barryk/tests/fork_loopLinux.1
7		intrepid.etnus.c	T	/home/barryk/tests/fork_loopLinux.2
8		intrepid.etnus.c	T	/home/barryk/tests/fork_loopLinux.1
9		intrepid.etnus.c	T	/home/barryk/tests/fork_loopLinux.1
10		intrepid.etnus.c	T	/home/barryk/tests/fork_loopLinux.1
11		intrepid.etnus.c	T	/home/barryk/tests/fork_loopLinux.3
12		intrepid.etnus.c	T	/home/barryk/tests/fork_loopLinux.2
13	1	<local>	B	mismatchLinux.1 (1 active threads)
13.1	1	<local>	B4	in main
14	2	<local>	B	mismatchLinux.2 (1 active threads)
15	3	<local>	B	mismatchLinux.3 (1 active threads)



MemoryScope 3.4.0-0

File Tools Window Help

Home Memory Reports Manage Processes **Memory Debugging Options** Tips

August 16, 2013

**Related Tasks**  
Load Memory Options  
Save Memory Options

**Other Tasks**  
Add Program  
Manage Processes  
Export Memory Data...

**Process Selection**

Process  
Parallel Job \_mpirun\_<cpi.mpt6>.0 (1)  
    MPI\_COMM\_WORLD  
        \_mpirun\_<cpi.mpt6>.0  
        \_mpirun\_<cpi.mpt6>.1  
        \_mpirun\_<cpi.mpt6>.2

**Memory Debugging Options**

Select your preferred level of debugging below or press *Advanced Options* for more control.

**Advanced Options**

☐ Enable memory debugging

**Levels of Debugging**

☐ Low  
Provides event notifications and leak detection. It allows the best performance for your process.

☒ Medium  
Adds corrupted memory detection by applying guard blocks. Performance may degrade slightly.

☐ High  
Provides memory over run alerts by monitoring Red Zone violations. Your memory consumption will increase significantly.

☐ Extreme  
Enables all options. There is a risk that performance may suffer and you will use more memory.

Yellow buttons ☐ mean:

- multiple processes are selected
- the settings can vary among selected processes
- you can modify the settings for all these processes by pressing the yellow buttons ☐

Note: You can select a single process to see its specific settings.

```
21 char *freeptr;  
22  
23 MPI_Init(&argc, &argv);  
24 MPI_Comm_size(MPI_COMM_WORLD, &numprocs);  
25 MPI_Comm_rank(MPI_COMM_WORLD, &myid);
```

Action Points Processes Threads

P- P+ T- T+



# Memory Debugging Options - Advanced



MemoryScope 3.2.1-0

File Tools Window Help

Home Memory Reports Manage Processes **Memory Debugging Options** Tips

June 12, 2011

**Related Tasks**  
Load Memory Options  
Save Memory Options

**Other Tasks**  
Add Program  
Manage Processes  
Export Memory Data

## Memory Debugging Options

Customize your options below or press *Basic Options* for predefined settings.

☒ **Enable memory debugging**

☐ **Halt execution on memory event or error**  
Use the **Advanced** button to control actions for individual events. [Advanced...](#)

**Memory Event Notification**

Select events to trigger: [All](#) [None](#)

Event	Description
<input checked="" type="checkbox"/> API usage error	Incorrect API or API instance used in operation
<input checked="" type="checkbox"/> Allocation failed	Error: An allocation call failed or the address returned is NULL which generally means out of memory
<input checked="" type="checkbox"/> Double allocation	Error: Allocator returned a block already in use; heap may be corrupted
<input checked="" type="checkbox"/> Double free	Error: Program attempted to free an already freed block
<input checked="" type="checkbox"/> Free interior pointer	Error: Program attempted to free a block incorrectly, via an address in the middle of the block
<input checked="" type="checkbox"/> Free notification	A block for which notification was requested is being freed
<input checked="" type="checkbox"/> Free unknown block	Error: Program attempted to free an address not in the heap
<input checked="" type="checkbox"/> Guard corruption error	Bounds error: The guard area around a block has been overwritten
<input checked="" type="checkbox"/> Invalid aligned allocation request	Error: Program supplied an invalid alignment argument to the heap manager
<input checked="" type="checkbox"/> Misaligned allocation	Error: Allocator returned a misaligned block; heap may be corrupted
<input checked="" type="checkbox"/> Realloc notification	A block for which notification was requested is being reallocated
<input checked="" type="checkbox"/> Realloc unknown block	Error: Program attempted to reallocate an address not in the heap
<input checked="" type="checkbox"/> Red Zone overrun error	Bounds error: Attempting to access memory beyond the end of an allocated block
<input checked="" type="checkbox"/> Red Zone overrun on deallocated block	Bounds error: Attempting to access memory beyond the end of a deallocated block
<input checked="" type="checkbox"/> Red Zone underrun error	Bounds error: Attempting to access memory before the start of an allocated block
<input checked="" type="checkbox"/> Red Zone underrun on deallocated block	Bounds error: Attempting to access memory before the start of a deallocated block
<input checked="" type="checkbox"/> Red Zone use-after-free error	Access error: Attempting to access a block after it has been deallocated
<input checked="" type="checkbox"/> Termination notification	The target is terminating, memory analysis can be performed

[Help](#) [OK](#) [Cancel](#)



# Memory Debugging Options - Advanced



MemoryScape 3.2.1-0

File Tools Window Help

Home Memory Reports Manage Processes **Memory Debugging Options** Tips

June 12, 2011

**Related Tasks**  
Load Memory Options  
Save Memory Options

**Other Tasks**  
Add Program  
Manage Processes  
Export Memory Data...

**Process Selection**

Process	Event
filterapp (0)	

**Memory Debugging Options**

Customize your options below or press *Basic Options* for predefined settings.

☒ **Enable memory debugging**

☒ **Guard allocated memory**

Pre-Guard Size: 8 bytes  
Pattern: 0x777777  
Post-Guard Size: 8 bytes  
Pattern: 0x999999  
Maximum Guard Size: 0 bytes

☒ **Use Red Zones to find memory access violations**

☒ **Paint memory**

☒ **Paint allocations** Pattern: 0xa110ca7f  
☒ **Paint deallocations** Pattern: 0xdea110cf

☒ **Hoard deallocated memory**

Maximum KB to hoard: 256 KB  
Maximum blocks to hoard: 39 blocks  
☒ Automatically release hoarded blocks when memory gets low

**Basic Options**

**Restore Defaults**



The screenshot shows the MemoryScope 3.4.0-0 application window. The title bar reads "MemoryScope 3.4.0-0". The menu bar includes "File", "Tools", "Window", and "Help". The toolbar contains icons for back, forward, home, and other navigation functions. The main menu bar has "Home", "Memory Reports", "Manage Processes", "Memory Debugging Options" (selected), and "Tips".

**Memory Debugging Options**

Select your preferred level of debugging below or press *Advanced Options* for more control.

**Advanced Options**

☒ Enable memory debugging

**Levels of Debugging**

- ☒ **Low**  
Provides event notifications and leak detection. It allows the best performance for your process.
- ☐ **Medium**  
Adds corrupted memory detection by applying guard blocks. Performance may degrade slightly.
- ☐ **High**  
Provides memory over run alerts by monitoring Red Zone violations. Your memory consumption will increase significantly.
- ☐ **Extreme**  
Enables all options. There is a risk that performance may suffer and you will use more memory.

**Yellow buttons ☒ mean:**

- multiple processes are selected
- the settings can vary among selected processes
- you can modify the settings for all these processes by pressing the yellow buttons ☒

Note: You can select a single process to see its specific settings.

**Process Selection**

Process ▾

Parallel Job \_mpirun\_<cpi.mpt6>.0 (

☒ MPI\_COMM\_WORLD

- ☒ \_mpirun\_<cpi.mpt6>.0
- ☒ \_mpirun\_<cpi.mpt6>.1
- ☒ \_mpirun\_<cpi.mpt6>.2

**Code Snippet:**

```
21 char *freeptr;  
22  
23 MPI_Init(&argc, &argv);  
24 MPI_Comm_size(MPI_COMM_WORLD, &numprocs);  
25 MPI_Comm_rank(MPI_COMM_WORLD, &myid);
```

**Action Points | Processes | Threads**

P- P+ T- T+



MemoryScape 3.4.0-0

File Tools Window Help

Home Memory Reports Manage Processes **Memory Debugging Options** Tips

August 16, 2013

**Related Tasks**

- Load Memory Options
- Save Memory Options

**Other Tasks**

- Add Program
- Manage Processes
- Export Memory Data...

**Process Selection**

Process ▾

- Parallel Job \_mpirun\_<cpi.mpt6>.0 (MPI\_COMM\_WORLD)
  - ▾ \_mpirun\_<cpi.mpt6>.0
  - ▾ \_mpirun\_<cpi.mpt6>.1
  - ▾ \_mpirun\_<cpi.mpt6>.2

**Memory Debugging Options**

Select your preferred level of debugging below or press *Advanced Options* for more control.

☐ Enable memory debugging

**Levels of Debugging**

- ☐ **Low**  
Provides event notifications and leak detection. It allows the best performance for your process.
- ☒ **Medium**  
Adds corrupted memory detection by applying guard blocks. Performance may degrade slightly.
- ☐ **High**  
Provides memory over run alerts by monitoring Red Zone violations. Your memory consumption will increase significantly.
- ☐ **Extreme**  
Enables all options. There is a risk that performance may suffer and you will use more memory.

**Advanced Options**

```
21     char *freeptr;  
22  
23     MPI_Init(&argc, &argv);  
24     MPI_Comm_size(MPI_COMM_WORLD, &numprocs);  
25     MPI_Comm_rank(MPI_COMM_WORLD, &myid);
```

Action Points Processes Threads

P- P+ T- T+



MemoryScape 3.4.0-0

File Tools Window Help

Home Memory Reports Manage Processes **Memory Debugging Options** Tips

August 16, 2013

**Related Tasks**

- Load Memory Options
- Save Memory Options

**Other Tasks**

- Add Program
- Manage Processes
- Export Memory Data...

**Memory Debugging Options**

Select your preferred level of debugging below or press *Advanced Options* for more control.

☐ Enable memory debugging

**Levels of Debugging**

- ☐ **Low**  
Provides event notifications and leak detection. It allows the best performance for your process.
- ☐ **Medium**  
Adds corrupted memory detection by applying guard blocks. Performance may degrade slightly.
- ☒ **High**  
Provides memory over run alerts by monitoring Red Zone violations. Your memory consumption will increase significantly.
- ☐ **Extreme**  
Enables all options. There is a risk that performance may suffer and you will use more memory.

**Process Selection**

Process ▾

- Parallel Job \_mpirun\_<cpi.mpt6>.0 (
- MPL\_COMM\_WORLD
- \_mpirun\_<cpi.mpt6>.0
- \_mpirun\_<cpi.mpt6>.1
- \_mpirun\_<cpi.mpt6>.2

Memory Debugging settings have been updated.

**Advanced Options**

```
21     char *freeptr;  
22  
23     MPI_Init(&argc, &argv);  
24     MPI_Comm_size(MPI_COMM_WORLD, &numprocs);  
25     MPI_Comm_rank(MPI_COMM_WORLD, &myid);
```

Action Points

Processes

Threads

P-

P+

T-

T+



MemoryScape 3.4.0-0

File Tools Window Help

Home Memory Reports Manage Processes **Memory Debugging Options** Tips

August 16, 2013

**Related Tasks**

- Load Memory Options
- Save Memory Options

**Other Tasks**

- Add Program
- Manage Processes
- Export Memory Data...

**Process Selection**

Process ▾

- Parallel Job \_mpirun\_<cpi.mpt6>.0
  - MPI\_COMM\_WORLD
    - \_mpirun\_<cpi.mpt6>.0
    - \_mpirun\_<cpi.mpt6>.1**
    - \_mpirun\_<cpi.mpt6>.2

Memory Debugging Options

Select your preferred level of debugging below or press *Advanced Options* for more control.

**Advanced Options**

☐ Enable memory debugging

**Levels of Debugging**

- ☐ **Low**  
Provides event notifications and leak detection. It allows the best performance for your process.
- ☒ **Medium**  
Adds corrupted memory detection by applying guard blocks. Performance may degrade slightly.
- ☐ **High**  
Provides memory over run alerts by monitoring Red Zone violations. Your memory consumption will increase significantly.
- ☐ **Extreme**  
Enables all options. There is a risk that performance may suffer and you will use more memory.

Memory Debugging settings have been updated.

```
21     char *freeptr;  
22  
23     MPI_Init(&argc, &argv);  
24     MPI_Comm_size(MPI_COMM_WORLD, &numprocs);  
25     MPI_Comm_rank(MPI_COMM_WORLD, &myid);
```

Action Points

Processes

Threads

P-

P+

T-

T+



2028809.pbsp11	ganatthe	normal	node_shuffle	1	1	00:10	0	10:41	--
2028810.pbsp11	ganatthe	normal	node_shuffle	1	1				
2028811.pbsp11	ganatthe	normal	node_shuffle	1	1				
2028812.pbsp11	ganatthe	normal	node_shuffle	1	1				
2028813.pbsp11	ganatthe	normal	node_shuffle	1	1				
2028814.pbsp11	ganatthe	normal	node_shuffle	1	1				
2028815.pbsp11	ganatthe	normal	node_shuffle	1	1				
[bash-3.2\$ pwd									

FileToolsWindowHelp

HomeMemory ReportsManage ProcessesMemory Debugging

August 16, 2013

Related Tasks

Load Memory Options

Save Memory Options

Other Tasks

Add Program

Manage Processes

Export Memory Data...

Memory Debugging

Select your preferred level of debugging

☐ Enable memory debugging

Levels of Debugging

☒ Low

Provides event notification

☐ Medium

Adds corrupted memory

☐ High

Provides memory overview

☐ Extreme

Enables all options.

Process Selection

Process

Parallel Job \_mpirun\_<cpi.mpt6>.0

MPI\_COMM\_WORLD

☒ \_mpirun\_<cpi.mpt6>.0

☒ \_mpirun\_<cpi.mpt6>.1

☒ \_mpirun\_<cpi.mpt6>.2

\_mpirun\_<cpi.mpt6>.0

FileEditViewGroupProcessThreadAction PointDebuggerToolsWindowHelp

Group (Control)

GoHaltKillRestartNext StepOutRun ToRecordGoBackPrevUnStepCallerBackToLive

Rank 0: \_mpirun\_<cpi.mpt6>.0 (Stopped) [M]

Thread 1 (46912521725664): cpi.mpt6 (Stopped)

Stack Trace

pthread\_barrier\_wait, FP=7fffffffe110

MPI\_SGI\_barrier, FP=7fffffffe0b0

MPI\_SGI\_init, FP=7fffffffe150

call\_init, FP=7fffffffe190

\_dl\_init, FP=7fffffffe1f0

Stack Frame

Registers for the frame:

%rax: 0xffffffffffffffe0 (-512)

%rdx: 0x00000009 (9)

%rcx: 0xffffffffffffffe0 (-1)

%rbx: 0x2aaaab20c298 (46912503857816)

%rsi: 0x00000000 (0)

%rdi: 0x2aaaad1e8100 (46912537264384)

%rbp: 0x7fffffffe110 (140737488347408)

%rsp: 0x7fffffffe0a8 (140737488347304)

%s0: 0x2aaaab20c298 (46912503857816)

Function main in cpi.c

3 #include <math.h>

4 #include <stdlib.h>

5 #include <string.h>

6

7 double f( double );

8 double f( double a )

9 {

10 return (4.0 / (1.0 + a\*a));

11 }

12

13 int main( int argc, char \*argv[] )

14 {

15 int done = 0, n, myid, numprocs, i;

16 double PI25DT = 3.141592653589793238462643;

17 double mypi, pi, h, sum, x;

18 double startvtime = 0.0, endvtime;

19 int namelen;

20 char processor\_name[MPI\_MAX\_PROCESSOR\_NAME];

21 char \*freeptr;

22

23 MPI\_Init(&argc, &argv);

24 MPI\_Comm\_size(MPI\_COMM\_WORLD, &numprocs);

25 MPI\_Comm\_rank(MPI\_COMM\_WORLD, &myid);

Action PointsProcessesThreads

P- P+ T- T+

...done

initially loaded at 0xffff000

1'...done

..done

initially loaded at 0xffff0

done

le

y loaded at 0xffff0

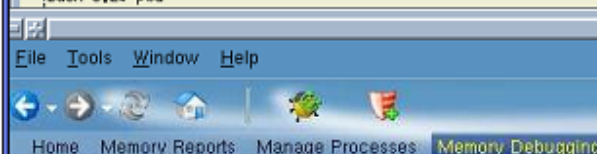
done

ions





2028809.pbsp11	ganatthe	normal	node_shuffle	1	1	00:10	Q	10:41	--
2028810.pbsp11	ganatthe	normal	node_shuffle	1	1				
2028811.pbsp11	ganatthe	normal	node_shuffle	1	1				
2028812.pbsp11	ganatthe	normal	node_shuffle	1	1				
2028813.pbsp11	ganatthe	normal	node_shuffle	1	1				
2028814.pbsp11	ganatthe	normal	node_shuffle	1	1				
2028815.pbsp11	ganatthe	normal	node_shuffle	1	1				
ibash-3.2\$ pud									



August 16, 2013

## Related Tasks

Load Memory Options  
Save Memory Options

## Other Tasks

Add Program  
Manage Processes  
Export Memory Data...

## Memory Debugging

Select your preferred level of

☐ Enable memory deb

## Levels of Debugging

- ☒ **Low**  
Provides event notifi
- ☐ **Medium**  
Adds corrupted men
- ☐ **High**  
Provides memory ov
- ☐ **Extreme**  
Enables all options.

## Process Selection

Process ▾  
Parallel Job \_mpirun\_<mpi.mpt6>.0 (  
MPI\_COMM\_WORLD  
\_mpirun\_<mpi.mpt6>.0  
\_mpirun\_<mpi.mpt6>.1  
\_mpirun\_<mpi.mpt6>.2

Stack Trace

Rank	Thread	Function	FP
0	_mpirun_<mpi.mpt6>.0	pthread_barrier_wait	FP=7fffffff110
		MPI_SGI_barasync	FP=7fffffff0b0
		MPI_SGI_init	FP=7fffffff150
		call_init	FP=7fffffff190
		_dl_init	FP=7fffffff1f0

Stack Frame

Registers for the frame:

%rax:	0xffffffffffffe00 (-512)
%rdx:	0x00000009 (9)
%rcx:	0xffffffffffff (-1)
%rbx:	0x2aaaab20c298 (46912503857816)
%rsi:	0x00000000 (0)
%rdi:	0x2aaaad1e8100 (46912537264384)
%rbp:	0x7fffffff110 (140737488347408)
%rsp:	0x7fffffff0a8 (140737488347304)

Function main in mpi.c

```

47     else
48     {
49         h = 1.0 / (double) n;
50         sum = 0.0;
51         x = f(h);
52         for (i = myid + 1; i <= n; i += numprocs)
53         {
54             /* comments to increase line number of malloc */
55             /*
56              *
57              */
58             x = h * ((double)i - 0.5);
59             sum += f(x);
60             freeptr = (char *) malloc ((unsigned long)150 *sizeof(char));
61             strcpy(freeptr, "Testing");
62         }
63         mypi = h + sum;
64         MPI_Reduce(&mypi, &pi, 1, MPI_DOUBLE, MPI_SUM, 0, MPI_COMM_WORLD);
65     }
66
67
68
69

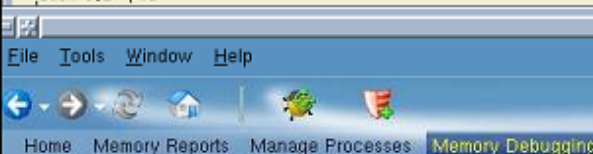
```

Action Points Processes Threads





2028809.pbsp11	ganatthe normal	node_shuffle	1	1	00:10 Q	10:41 --
2028810.pbsp11	ganatthe normal	node_shuffle	1	1		
2028811.pbsp11	ganatthe normal	node_shuffle	1	1		
2028812.pbsp11	ganatthe normal	node_shuffle	1	1		
2028813.pbsp11	ganatthe normal	node_shuffle	1	1		
2028814.pbsp11	ganatthe normal	node_shuffle	1	1		
2028815.pbsp11	ganatthe normal	node_shuffle	1	1		
[bash-3.2\$ pwd						



August 16, 2013

Related Tasks

- Load Memory Options
- Save Memory Options

Other Tasks

- Add Program
- Manage Processes
- Export Memory Data...

Memory Debugging

Select your preferred level of debugging:

- ☐ Enable memory debugging

Levels of Debugging

- ☐ Low  
Provides event notification
- ☐ Medium  
Adds corrupted memory
- ☐ High  
Provides memory overflow
- ☐ Extreme  
Enables all options.

Process Selection

Process

Parallel Job \_mpirun\_<mpi.mpt6>.0 (MPI\_COMM\_WORLD)

- ☐ \_mpirun\_<mpi.mpt6>.0
- ☐ \_mpirun\_<mpi.mpt6>.1
- ☐ \_mpirun\_<mpi.mpt6>.2

Stack Trace

Rank 0: _mpirun_<mpi.mpt6>.0 (Stopped) [M]	Thread 1 (46912521725664): cpi.mpt6 (Stopped)
pthread_barrier_wait, FP=7fffffff110	
MPI_SGI_bar_sync, FP=7fffffff0b0	
MPI_SGI_init, FP=7fffffff150	
call_init, FP=7fffffff190	
_dl_init, FP=7fffffff1f0	

Stack Frame

Registers for the frame:

%rax:	0xffffffffffffe00 (-512)
%rdx:	0x00000009 (9)
%rcx:	0xffffffffffff (-1)
%rbx:	0x2aaaab20c298 (46912503857816)
%rsi:	0x00000000 (0)
%rdi:	0x2aaaad1e8100 (46912537264384)
%rbp:	0x7fffffff110 (140737488347408)
%rsp:	0x7fffffff0a8 (140737488347304)
%r8:	0x2aaaab20c298 (46912503857816)

Function main in cpi.c

```

47     else
48     {
49         h = 1.0 / (double) n;
50         sum = 0.0;
51         x = f(h);
52         for (i = myid + 1; i <= n; i += numprocs)
53         {
54             /* comments to increase line number of malloc */
55             /*
56              *
57              */
58             x = h * ((double)i - 0.5);
59             sum += f(x);
60             freeptr = (char *) malloc ((unsigned long)150 * sizeof(char));
61             strcpy(freeptr, "Testing");
62         }
63         mypi = h * sum;
64         MPI_Reduce(&mypi, &pi, 1, MPI_DOUBLE, MPI_SUM, 0, MPI_COMM_WORLD);
65     }
66
67
68
69

```

Action Points Processes Threads

STOP 1 cpi.c#63 main+0x182

initially loaded at 0xffff000

1'...done

..done

initially loaded at 0xffff

done

se

y loaded at 0xffff0

done

ions



Process	Command	State	PPID	UID	Time	Time	Time
2028809.pbsp11	ganatthe normal	node_shuffle	1	1	00:10	0	10:41
2028810.pbsp11	ganatthe normal	node_shuffle	1	1			
2028811.pbsp11	ganatthe normal	node_shuffle	1	1			
2028812.pbsp11	ganatthe normal	node_shuffle	1	1			
2028813.pbsp11	ganatthe normal	node_shuffle	1	1			
2028814.pbsp11	ganatthe normal	node_shuffle	1	1			
2028815.pbsp11	ganatthe normal	node_shuffle	1	1			
ibash-3.2\$	pwd						

File Tools Window Help

Home Memory Reports Manage Processes **Memory Debugging**

August 16, 2013

**Related Tasks**

- Load Memory Options
- Save Memory Options

**Other Tasks**

- Add Program
- Manage Processes
- Export Memory Data...

Process Selection

Process ▾

Parallel Job \_mpirun\_<mpi.mpt6>.0 (

- ▾ MPI\_COMM\_WORLD
  - ▾ \_mpirun\_<mpi.mpt6>.0
  - ▾ \_mpirun\_<mpi.mpt6>.1**
  - ▾ \_mpirun\_<mpi.mpt6>.2

### Memory Debugging

Select your preferred level of

☐ Enable memory deb

Levels of Debugging

- ☐ **Low**  
Provides event notifi
- ☐ **Medium**  
Adds corrupted men
- ☐ **High**  
Provides memory ov
- ☐ **Extreme**  
Enables all options.

\_mpirun\_<mpi.mpt6>.0

File Edit View Group Process Thread Action Point Debug Tools Window Help

Group (Control)

Rank 0: \_mpirun\_<mpi.mpt6>.0 (At Breakpoint 1) [M]

Thread 1 (46912521725664): mpi.mpt6 (At Breakpoint 1)

Stack Trace

Frame	Address	Function
0	FP=7fffffffe130	main.
1	FP=7fffffffe1e0	_libc_start_main.
2	FP=7fffffffe1f0	_start.

Stack Frame

Function "main":

argc: 0x00000001 (1)

argv: 0x7fffffffe208 -> 0x7fffffffe5e

Local variables:

done: 0x00000000 (0)

n: 0x00000064 (100)

myid: 0x00000000 (0)

numprocs: 0x00000008 (8)

i: 0x00000001 (1)

PI25DT: 3.14159265358979

mypi: 2.07395319538588e-317 <denormal:

Function main in mpi.c

```

51         x = f(h);
52         for (i = myid + 1; i <= n; i += numprocs)
53         {
54             /* comments to increase line number of malloc */
55             /*
56              *
57              */
58             /*
59              */
60             x = h + ((double)i - 0.5);
61             sum += f(x);
62             freeptr = (char *) malloc((unsigned long)150 * sizeof(char));
63             strcpy(freeptr, "Testing");
64         }
65         mypi = h * sum;
66         MPI_Reduce(&mypi, Spi, 1, MPI_DOUBLE, MPI_SUM, 0, MPI_COMM_WORLD);
67         if (myid == 0)
68         {
69             printf("pi is approximately %16f. Error is %16f\n",
70                   pi, fabs(pi - PI25DT));
71         }
72     }
73

```

Action Points Processes Threads

1 mpi.c#63 main+0x182

2,so'...done

o'...done

initially loaded at 0xff0

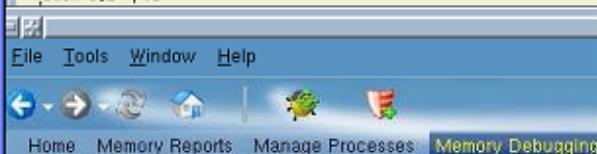
so'...done

tions





2028809.pbsp11	ganatthe	normal	node_shuffle	1	1	00:10	Q	10:41	--
2028810.pbsp11	ganatthe	normal	node_shuffle	1	1				
2028811.pbsp11	ganatthe	normal	node_shuffle	1	1				
2028812.pbsp11	ganatthe	normal	node_shuffle	1	1				
2028813.pbsp11	ganatthe	normal	node_shuffle	1	1				
2028814.pbsp11	ganatthe	normal	node_shuffle	1	1				
2028815.pbsp11	ganatthe	normal	node_shuffle	1	1				
[bash-3.2\$ pwd									



August 16, 2013

## Related Tasks

Load Memory Options  
Save Memory Options

## Other Tasks

Add Program  
Manage Processes  
Export Memory Data...

## Memory Debugging

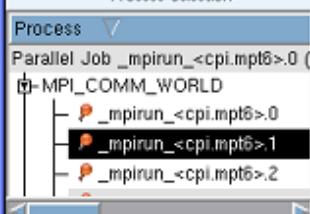
Select your preferred level of

☐ Enable memory deb

## Levels of Debugging

- ☐ Low  
Provides event notifi
- ☐ Medium  
Adds corrupted men
- ☐ High  
Provides memory ov
- ☐ Extreme  
Enables all options.

## Process Selection



File Edit View Group Process Thread Action Point Debug Tools Window Help

Group (Control)

Go Halt Kill Restart Next Step Out Run To Record GoBack Prev UnStep Caller BackTo Live

Rank 0: \_mpirun\_<mpi.mpt6>.0 (At Breakpoint 1) [M]

Thread 1 (46912521725664): cpi.mpt6 (At Breakpoint 1)

Stack Trace

C main, FP=7fffffff130  
\_libc\_start\_main, FP=7fffffff1e0  
\_start, FP=7fffffff1f0

Stack Frame

Function "main":  
argv: 0x00000001 (1)  
argv: 0x7fffffff208 -> 0x7fffffff5e  
Local variables:  
done: 0x00000000 (0)  
n: 0x00000064 (100)  
myid: 0x00000000 (0)  
numprocs: 0x00000008 (8)  
i: 0x00000001 (1)  
PI25DT: 3.14159265358979  
mypi: 2.07395319538588e-317 <denormal:  
4.84666440941947e-394 <denormal>

Function main in cpi.c

```

51         x = f(h);
52         for (i = myid + 1; i <= n; i += numprocs)
53         {
54             /* comments to increase line number of malloc */
55             /*
56              *
57              */
58             /*
59              *
60              */
61             x = h + ((double)i - 0.5);
62             sum += f(x);
63             freeptr = (char *) malloc ((unsigned long)150 * sizeof(char));
64             strcpy(freeptr, "Memory");
65             freeptr: 0x006091a0 (Allocated) -> "H325k253252"
66         }
67         mypi = h * sum;
68         MPI_Reduce(&mypi, Spi, 1, MPI_DOUBLE, MPI_SUM, 0, MPI_COMM_WORLD);
69         if (myid == 0)
70         {
71             printf("pi is approximately %.16f. Error is %.16f\n",
72                   pi, fabs(pi - PI25DT));
73         }

```

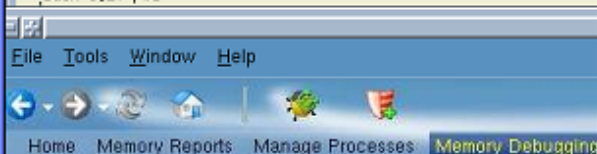
Action Points Processes Threads

1 cpi.c#63 main=0x182





2028809.pbsp11	ganatthe	normal	node_shuffle	1	1	00:10	Q	10:41	--
2028810.pbsp11	ganatthe	normal	node_shuffle	1	1				
2028811.pbsp11	ganatthe	normal	node_shuffle	1	1				
2028812.pbsp11	ganatthe	normal	node_shuffle	1	1				
2028813.pbsp11	ganatthe	normal	node_shuffle	1	1				
2028814.pbsp11	ganatthe	normal	node_shuffle	1	1				
2028815.pbsp11	ganatthe	normal	node_shuffle	1	1				
[bash-3.2\$ pwd									



August 16, 2013

**Related Tasks**

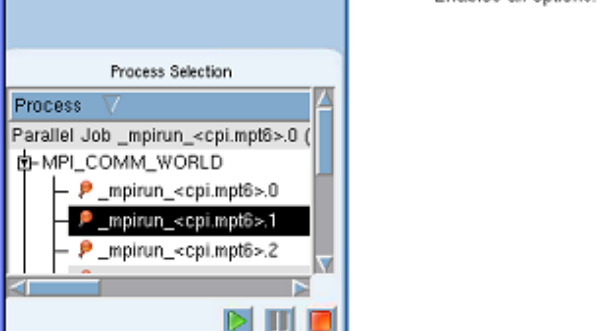
- Load Memory Options
- Save Memory Options

**Other Tasks**

- Add Program
- Manage Processes
- Export Memory Data...

**Levels of Debugging**

- ☐ Low  
Provides event notification
- ☐ Medium  
Adds corrupted memory
- ☐ High  
Provides memory overview
- ☐ Extreme  
Enables all options.



File Edit View Group Process Thread Action Point Debug Tools Window Help

Group (Control)

Halt

Kill

Restart

Next Step

Out

Run To

Record

GoBack

Prev

UnStep

Caller

BackTo

Live

Rank 0: \_mpirun\_<mpi.mpt6> 0 (At Breakpoint 1) [M]

Thread 1 (46912521725664): cpi.mpt6 (At Breakpoint 1)

Stack Trace

C main, FP=7fffffffe130

\_libc\_start\_main, FP=7fffffffe1e0

\_start, FP=7fffffffe1f0

Stack Frame

Function "main":

argv: 0x00000001 (1)

argc: 0x7fffffffe208 -> 0x7fffffffe5e

Local variables:

done: 0x00000000 (0)

n: 0x00000064 (100)

myid: 0x00000000 (0)

numprocs: 0x00000008 (8)

i: 0x00000001 (1)

PI25DT: 3.14159265358979

mypi: 2.07395319538588e-317 <denormal: 4.8466644041047e-394 <denormal>

Function main in cpi.c

51 x = f(h);

52 for (i = myid + 1; i <= n; i += numprocs)

53 {

54 /\* comments to increase line number of malloc \*/

55 /\*

56 \*

57 \*

58 \*/

59

60 x = h + ((double)i - 0.5);

61 sum += f(x);

62 freeptr = (char \*) malloc ((unsigned long)150 \*sizeof(char));

63 strcpy(freeptr, "Testing");

64

65 }

66 mypi = h \* sum;

67

68 MPI\_Reduce(&mypi, Spi, 1, MPI\_DOUBLE, MPI\_SUM, 0, MPI\_COMM\_WORLD);

69

70 if (myid == 0)

71 {

72 printf("pi is approximately %.16f. Error is %.16f\n",

73 pi, fabs(pi - PI25DT));

Action Points Processes Threads

1 cpi.c#63 main=0x182



The screenshot displays a Linux desktop environment accessed via TightVNC. The desktop features a taskbar with icons for 'fwmm', 'openSUSE', 'Graphics', 'Tools', and 'Shells'. A terminal window in the background shows a list of processes with columns for PID, username, session ID, and process name, all running 'node\_shuffle'. In the foreground, the 'MemoryScope 3.4.0-0' application is open, showing the 'Memory Debugging Options' dialog. This dialog includes a 'Process Selection' tree on the left, listing processes like 'Parallel Job \_mpirun\_<mpi.mpt6>.0' and its sub-processes. The main area of the dialog allows users to select a debugging level: Low, Medium, High, or Extreme, each with a brief description of its capabilities and performance impact. A 'Related Tasks' sidebar on the left offers options like 'Load Memory Options' and 'Save Memory Options'. The top of the desktop shows a clock and system status indicators.

Process Selection

- Process
- Parallel Job \_mpirun\_<mpi.mpt6>.0
  - MPI\_COMM\_WORLD
    - \_mpirun\_<mpi.mpt6>.0
    - \_mpirun\_<mpi.mpt6>.1
    - \_mpirun\_<mpi.mpt6>.2

Memory Debugging Options

Select your preferred level of debugging below or press *Advanced Options* for more control.

☐ Enable memory debugging

Levels of Debugging

- ☐ Low  
Provides event notifications and leak detection. It allows the best performance for your process.
- ☐ Medium  
Adds corrupted memory detection by applying guard blocks. Performance may degrade slightly.
- ☐ High  
Provides memory over run alerts by monitoring Red Zone violations. Your memory consumption will increase significantly.
- ☐ Extreme  
Enables all options. There is a risk that performance may suffer and you will use more memory.

Related Tasks

- Load Memory Options
- Save Memory Options

Other Tasks

- Add Program
- Manage Processes
- Export Memory Data...



The screenshot displays a Linux desktop environment accessed via TightVNC. The desktop features a taskbar with icons for 'fvm2', 'openSUSE', 'Graphics', 'Tools', and 'Shells'. A terminal window in the background shows a list of processes with columns for PID, PPID, USER, NAME, and STATE. The foreground window is 'MemoryScape 3.4.0-0', which is currently displaying the 'Memory Reports' section. This section includes a 'Summary' tab and a 'Create Reports' button. Below the button, there are several report categories with descriptions:

- Heap Status Reports**: Use heap status reports to analyze where your program is using memory, is generating leaks and how memory is physically being allocated on the heap. View a graphical depiction of the heap with the [Heap Graphical Report](#), view memory data broken down by source code with the [Source Report](#) or view memory allocations broken down by unique call stack traces with the [Backtrace Report](#).
- Leak Detection Reports**: Finding and fixing memory leaks can be quite challenging. Use the leak detection [Source Report](#) to view memory leaks broken down by source code or use the [Backtrace Report](#) to see leaks broken down by unique call stack traces.
- Memory Usage Reports**: Memory usage reports provide high level memory region information about your process. Use the [Chart Report](#) to see memory regions broken out using graphs and charts or view process level or library level memory region details with the [High Level Process Report](#) or the Detailed [Program and Library Report](#).
- Memory Corruption Reports**: If you configured MemoryScape to use guard blocks, the [Corrupted Memory Report](#) will show you if any memory blocks were corrupted.
- Memory Comparison Reports**: Compare how heap usage has changed using the [Memory Comparison Report](#) by comparing two different memory data sources. Memory data sources can include live processes, memory debugging data files and even core files containing memory data.

A 'Process Selection' dialog box is open in the bottom left corner, showing a tree view of processes. The selected process is 'Parallel Job \_mpirun\_<mpi.mpt6>.0', which contains sub-processes: 'MPI\_COMM\_WORLD', '\_mpirun\_<mpi.mpt6>.0', '\_mpirun\_<mpi.mpt6>.1', and '\_mpirun\_<mpi.mpt6>.2'.



The screenshot shows a TightVNC session of a Linux desktop. At the top, there's a taskbar with icons for 'Fvwm2', 'openSUSE', 'Graphics', 'Tools', and 'Shells'. Below the taskbar, a terminal window displays a list of processes with columns for PID, user, group, and other details. A file manager window is open, showing a directory listing. The main window is 'MemoryScope 3.4.0-0', which is displaying the 'Create Leak Detection Reports' page. The page has a sidebar with navigation links like 'Summary', 'Leak Detection', 'Heap Status', 'Memory Usage', 'Corrupted Memory', and 'Memory Comparisons'. The main content area explains how to find and fix memory leaks and provides links for 'Source report' and 'Backtrace report'. A 'Process Selection' dialog is open in the bottom left, showing a tree view of processes, with 'MPI\_COMM\_WORLD' selected.

2028809.pbsp11 gamatthe normal node\_shuffle 1 1 00:10 Q 10:41 --

2028810.pbsp11 gamatthe normal node\_shuffle 1 1

2028811.pbsp11 gamatthe normal node\_shuffle 1 1

2028812.pbsp11 gamatthe normal node\_shuffle 1 1

2028813.pbsp11 gamatthe normal node\_shuffle 1 1

2028814.pbsp11 gamatthe normal node\_shuffle 1 1

2028815.pbsp11 gamatthe normal node\_shuffle 1 1

ibash-  
/u/pn  
bash-  
bash:  
bash:  
bash-  
exit  
Znodu  
No No  
Znodu  
nodue  
Z nod  
Znodu  
Zuhic  
/nasa  
Zsete  
Znpic

File Edit View Group Process Thread Action Point Debug Tools Window Help

Group (Control)

MemoryScope 3.4.0-0

File Tools Window Help

Home Memory Reports Manage Processes Memory Debugging Options Tips

Summary | Leak Detection | Heap Status | Memory Usage | Corrupted Memory | Memory Comparisons

August 16, 2013

Save Data  
Export Memory Data...

Leak Detection Reports  
Source Report  
Backtrace Report

Other Reports Categories  
Heap Status Reports  
Memory Usage Reports  
Corrupted Memory Report  
Compare Memory Usage

Other  
Manage Filters

Process Selection

Process  
Parallel Job \_mpirun\_<mpi.mpt6>.0  
MPI\_COMM\_WORLD  
\_mpirun\_<mpi.mpt6>.0  
\_mpirun\_<mpi.mpt6>.1  
\_mpirun\_<mpi.mpt6>.2

Create Leak Detection Reports

Finding and fixing memory leaks can be quite the challenge. These reports provide several ways for you to examine the instances of leaked memory that were found by MemoryScope.

Leak detection reports

Source report  
Analyzes memory usage and displays leaks according to the point of allocation in source code.

Backtrace report  
Analyzes memory usage and displays leaks according to their point of allocation by unique call stack or backtrace ids.



The screenshot displays a Linux desktop environment accessed via TightVNC. The desktop features a taskbar with icons for 'Fvwm2', 'openSUSE', 'Graphics', 'Tools', and 'Shells'. A terminal window in the background shows a list of processes with columns for PID, user, session, and process name. The foreground application is 'MemoryScape 3.4.0-0', which is displaying a 'Leak Detection Backtrace Report' for the process '\_mpirun\_<mpi.mpt6>.1'. The report includes a table with columns for Process, Total Bytes, Count, Function, Line #, and Source Information. The table lists three memory leaks at addresses 0x220, 0x150, and 0x121. A 'Process Selection' dialog is open in the bottom-left corner, showing a tree view of the process hierarchy with '\_mpirun\_<mpi.mpt6>.1' selected.

**Terminal Window Output:**

PID	USER	SESSION	PROCESS
2028809	pbsp11	ganatthe	normal
2028810	pbsp11	ganatthe	normal
2028811	pbsp11	ganatthe	normal
2028812	pbsp11	ganatthe	normal
2028813	pbsp11	ganatthe	normal
2028814	pbsp11	ganatthe	normal
2028815	pbsp11	ganatthe	normal

**MemoryScape 3.4.0-0 - Leak Detection Backtrace Report**

Options: ☒ Relative to Baseline ☒ Enable Filtering

Process	Total Bytes	Count	Function	Line #	Source Information
Process 3: _mpirun_<mpi.mpt6>.1	468	4			
0x220	300	2			
0x150	72	1			
0x121	96	1			

**Process Selection Dialog:**

- Process
- Parallel Job \_mpirun\_<mpi.mpt6>.0
- MPI\_COMM\_WORLD
- \_mpirun\_<mpi.mpt6>.0
- \_mpirun\_<mpi.mpt6>.1**
- \_mpirun\_<mpi.mpt6>.2



Taskbar icons: Fvwm2, openSUSE, Graphics, Tools, Shells.

Terminal window output:

```
2028809.pbsp11 ganatthe normal node_shuffle 1 1 00:10 Q 10:41 --
2028810.pbsp11 ganatthe normal node_shuffle 1 1
2028811.pbsp11 ganatthe normal node_shuffle 1 1
2028812.pbsp11 ganatthe normal node_shuffle 1 1
2028813.pbsp11 ganatthe normal node_shuffle 1 1
2028814.pbsp11 ganatthe normal node_shuffle 1 1
2028815.pbsp11 ganatthe normal node_shuffle 1 1
ibash-7.2
bash-7.2
bash-7.2
bash-7.2
bash-7.2
exit
Znodu
No No
Znodu
nodue
Z nod
Znodu
Zuhic
/nasa
Zsete
Zapic
```

MemoryScope 3.4.0-0

File Tools Window Help

Home Memory Reports Manage Processes Memory Debugging Options Tips

Summary | Leak Detection | Heap Status | Memory Usage | Corrupted Memory | Memory Comparisons

August 16, 2013

Save Data  
Save Report...  
Export Memory Data...

Leak Detection Reports  
Source Report

Other Reports Categories  
Heap Status Reports  
Memory Usage Reports  
Corrupted Memory Report  
Compare Memory Usage

Other  
Manage Filters

Process Selection

Process: Parallel Job \_mpirun\_<cpi.mpt6>.0  
MPI\_COMM\_WORLD  
\_mpirun\_<cpi.mpt6>.0  
\_mpirun\_<cpi.mpt6>.1  
\_mpirun\_<cpi.mpt6>.2

### Leak Detection Backtrace Report

Options  
☒ Relative to Baseline ☐ Enable Filtering

Process	Total Bytes	Count	Function	Line #	Source Information
Process 3: _mpirun_<cpi.mpt6>.1	468	4			
220	300	2	malloc	166	malloc_wrappers_dlopen.c
			main	62	cpi.c
			_libc_start_main		libc.so.6
			_start		cpi.mpt6
150	72	1			
121	96	1			

Source

```
59
60 x = h * ((double)i - 0.5);
61 sum += f(x);
62 freeptr = (char *) malloc ((unsigned long)150 * sizeof(char));
63 strcpy(freeptr, "Testing");
64
```

/u/pmthomp/cpi.c



**Process List (from terminal):**

Process ID	User	Group	Process Name	State	PPID	Time	Time	Time
2028809.pbsp11	ganatthe	normal	node_shuffle	1	1	00:10	0	10:41
2028810.pbsp11	ganatthe	normal	node_shuffle	1	1	00:10	0	10:41
2028811.pbsp11	ganatthe	normal	node_shuffle	1	1	00:10	0	10:41
2028812.pbsp11	ganatthe	normal	node_shuffle	1	1	00:10	0	10:41
2028813.pbsp11	ganatthe	normal	node_shuffle	1	1	00:10	0	10:41
2028814.pbsp11	ganatthe	normal	node_shuffle	1	1	00:10	0	10:41
2028815.pbsp11	ganatthe	normal	node_shuffle	1	1	00:10	0	10:41

**MemoryScope 3.4.0-0: Leak Detection Backtrace Report**

Options: ☒ Relative to Baseline ☒ Enable Filtering

Process	Total Bytes	Count	Function	Line #	Source Information
Process 3: _mpirun_<mpi.mpt6>.1	468	4			
├─220	300	2	malloc	166	malloc_wrappers_dlopen.c
│			main	62	cpi.c
│			_libc_start_main		libc.so.6
└─150	72	1	_start		cpi.mpt6
└─121	96	1			

**Process Selection:**

- Parallel Job \_mpirun\_<mpi.mpt6>.0
- MPI\_COMM\_WORLD
  - \_mpirun\_<mpi.mpt6>.0
  - \_mpirun\_<mpi.mpt6>.1**
  - \_mpirun\_<mpi.mpt6>.2

**Source Code (Line 62):**

```

59
60     x = h * ((double)i - 0.5);
61     sum += f(x);
62     freeptr = (char *) malloc ((unsigned long)150 * sizeof(char));
63     strcpy(freeptr, "Testing");
64
  
```





# Example using *mpi-sgi/mpt.2.08r7*



MemoryScope 3.4.0-0

File Tools Window Help

Home **Memory Reports** Manage Processes Memory Debugging Options Tips

Summary | **Leak Detection** | Heap Status | Memory Usage | Corrupted Memory | Memory Comparisons

August 7, 2013

Save Data  
Save Report...  
Export Memory Data...

Leak Detection Reports  
Backtrace Report

Other Reports Categories  
Heap Status Reports  
Memory Usage Reports  
Corrupted Memory Report  
Compare Memory Usage

Other  
Manage Filters

Process Selection

Process  
Parallel Job \_mpirun\_<cpi\_mem>.0  
  MPI\_COMM\_WORLD  
    \_mpirun\_<cpi\_mem>.0  
    \_mpirun\_<cpi\_mem>.1  
    \_mpirun\_<cpi\_mem>.2

Options

☒ Relative to Baseline ☐ Enable Filtering

Process	Bytes	Count	Begin Address	End Address	Backtrace ID	Allocator
Process 2: _mpirun_<cpi_mem>.0	1156	8				
libmpi.so	1156	8				
malloc_intercept.c	1060	7				
malloc	1060	7				
Line 57	1060	7				
Block 1.4	160	1	0x00603720	0x006037bf	1	
Block 1.108	150	1	0x00da0c90	0x00da0d25	1	
Block 1.107	150	1	0x00da0bd0	0x00da0c65	1	
Block 1.100	150	1	0x00da07f0	0x00da07f5	1	

Backtrace

ID	Function	Line #	Source Information
1			

Source

1	No source available.
---	----------------------

Action Points Processes Threads

P- P+ T- T+

STOP	1	cpi.c#56	main+0x174
→	2	cpi.c#62	main+0x1b2

Help

KTo Live

normal:

Test:

char))



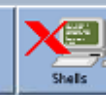
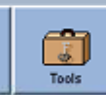


# Going back to Example using *mpi-sgi/mpt.2.06r6*







[illegible]

Process Thread

```
2,so'...done
o'...done
initially loaded at 0xff0
```

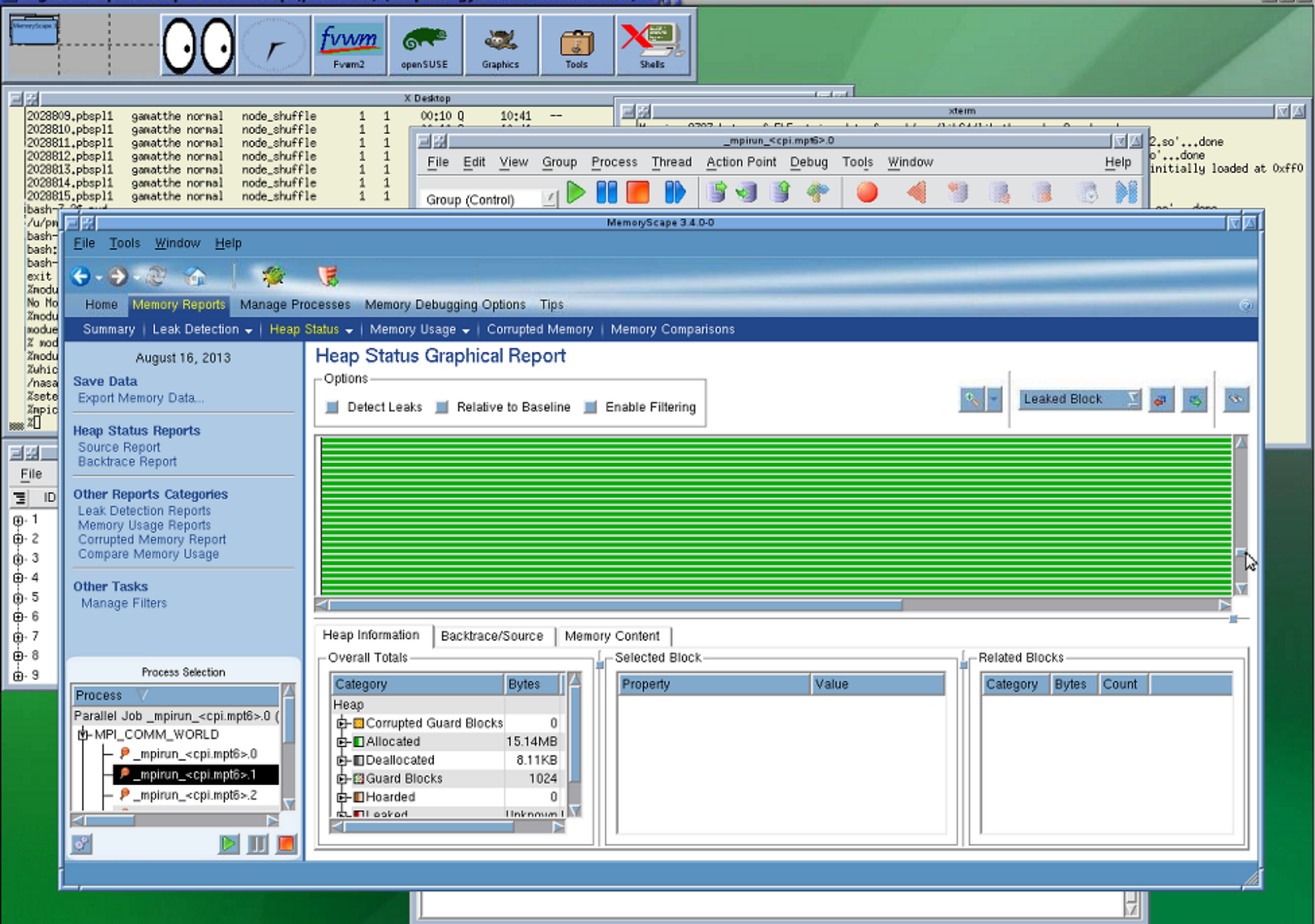
- Heap Status Reports
  - Source Report
  - Backtrace Report
- Other Reports Categories
  - Leak Detection Reports
  - Memory Usage Reports
  - Corrupted Memory Report
  - Compare Memory Usage
- Other Tasks
  - Manage Filters

Selected Block

Property	Value

Related Blocks			
Category	Bytes	Count	





2028809.pbsp11 gawatthe normal node\_shuffle 1 1 00:10 Q 10:41 --

2028810.pbsp11 gawatthe normal node\_shuffle 1 1

2028811.pbsp11 gawatthe normal node\_shuffle 1 1

2028812.pbsp11 gawatthe normal node\_shuffle 1 1

2028813.pbsp11 gawatthe normal node\_shuffle 1 1

2028814.pbsp11 gawatthe normal node\_shuffle 1 1

2028815.pbsp11 gawatthe normal node\_shuffle 1 1

bash-3.2\$

bash-3.2\$ /u/pn

bash-3.2\$

bash-3.2\$

bash-3.2\$ exit

2nodu

No No

2nodu

nodue

Z nod

2nodu

Zuhic

/nasa

Zsete

Znpic

File

ID

1

2

3

4

5

6

7

8

9

Process Selection

Process

Parallel Job \_mpirun\_<mpi.mpt6>.0 (

MPI\_COMM\_WORLD

\_mpirun\_<mpi.mpt6>.0

\_mpirun\_<mpi.mpt6>.1

\_mpirun\_<mpi.mpt6>.2

MemoryScope 3.4.0-0

File Tools Window Help

Home Memory Reports Manage Processes Memory Debugging Options Tips

Summary Leak Detection Heap Status Memory Usage Corrupted Memory Memory Comparisons

August 16, 2013

Save Data

Export Memory Data...

Heap Status Reports

Source Report

Backtrace Report

Other Reports Categories

Leak Detection Reports

Memory Usage Reports

Corrupted Memory Report

Compare Memory Usage

Other Tasks

Manage Filters

Options

Detect Leaks Relative to Baseline Enable Filtering

Leaked Block

Heap Information Backtrace/Source Memory Content

Overall Totals

Category	Bytes
Heap	
Corrupted Guard Blocks	0
Allocated	15.14MB
Deallocated	8.11KB
Guard Blocks	1024
Hoarded	0
Leaked	Unknown

Selected Block

Property	Value
----------	-------

Related Blocks

Category	Bytes	Count
----------	-------	-------



The screenshot displays a Linux desktop environment accessed via TightVNC. The desktop features a taskbar with icons for 'fwvm', 'openSUSE', 'Graphics', 'Tools', and 'Shell'. A terminal window in the background shows a list of processes, including several instances of 'node\_shuffle'. The foreground is dominated by the 'MemoryScope 3.4.0-0' application, which is displaying a 'Heap Status Graphical Report' for the process '\_mpirun\_<mpi.mpt6>.1'. The report includes a memory usage bar chart and a table of heap information.

**MemoryScope 3.4.0-0**

File Tools Window Help

Home **Memory Reports** Manage Processes Memory Debugging Options Tips

Summary | Leak Detection | **Heap Status** | Memory Usage | Corrupted Memory | Memory Comparisons

August 16, 2013

Save Data  
Export Memory Data...

**Heap Status Reports**  
Source Report  
Backtrace Report

**Other Reports Categories**  
Leak Detection Reports  
Memory Usage Reports  
Corrupted Memory Report  
Compare Memory Usage

**Other Tasks**  
Manage Filters

Process Selection

Process ▾  
Parallel Job \_mpirun\_<mpi.mpt6>.0 (1)  
MPI\_COMM\_WORLD  
\_mpirun\_<mpi.mpt6>.0  
**\_mpirun\_<mpi.mpt6>.1**  
\_mpirun\_<mpi.mpt6>.2

**Heap Status Graphical Report**

Options  
☒ Detect Leaks ☒ Relative to Baseline ☒ Enable Filtering

Process 3: \_mpirun\_<mpi.mpt6>.1

0x006030b0 - 0x01328cd0 (13.15MB)

2.00KB

Heap Information Backtrace/Source Memory Content

Backtrace

ID	Total Bytes	Count	Function	Line #
8	1176	1		
9	27	1		
10	88	1		
11	264	1		
12	168	1		
14	174	14		
15	56	1		
16	11	1		

Source



Taskbar icons:

Terminal window (xterm) output:

```
2028809.pbsp11 ganatthe normal node_shuffle 1 1 00:10 Q 10:41 --
2028810.pbsp11 ganatthe normal node_shuffle 1 1
2028811.pbsp11 ganatthe normal node_shuffle 1 1
2028812.pbsp11 ganatthe normal node_shuffle 1 1
2028813.pbsp11 ganatthe normal node_shuffle 1 1
2028814.pbsp11 ganatthe normal node_shuffle 1 1
2028815.pbsp11 ganatthe normal node_shuffle 1 1
ibash:
/u/pn
bash:
bash:
bash:
exit
Znodu
No No
Znodu
nodue
Z nod
Znodu
Zuhic
/nasa
Zsete
Znpic
```

MemoryScope 3.4.0-D-0

File Tools Window Help

Home **Memory Reports** Manage Processes Memory Debugging Options Tips

Summary | Leak Detection | **Heap Status** | Memory Usage | Corrupted Memory | Memory Comparisons

August 16, 2013

Save Data  
Export Memory Data...

Heap Status Reports  
Source Report  
Backtrace Report

Other Reports Categories  
Leak Detection Reports  
Memory Usage Reports  
Corrupted Memory Report  
Compare Memory Usage

Other Tasks  
Manage Filters

Process Selection

Process **\_mpirun\_<cpi.mpt6>.1**

Parallel Job \_mpirun\_<cpi.mpt6>.0  
MPI\_COMM\_WORLD  
\_mpirun\_<cpi.mpt6>.0  
**\_mpirun\_<cpi.mpt6>.1**  
\_mpirun\_<cpi.mpt6>.2

Heap Status Graphical Report

Options  
☒ Detect Leaks ☒ Relative to Baseline ☒ Enable Filtering

Process 3: \_mpirun\_<cpi.mpt6>.1

0x006030b0 - 0x01328cd0 (13.15MB)

2.00KB

Leaked Block

Heap Information Backtrace/Source Memory Content

Backtrace

ID	Total Bytes	Count	Function	Line
			dlopen_doit	
			_dl_catch_error	
			_dlerror_run	
			dlopen	
			<b>MPI_SGI_load_ib fu...</b>	
9	27	1		
10	88	1		

Source

1 No source available.



**MemoryScape 3.4.0-0**

File Tools Window Help

Home **Memory Reports** Manage Processes Memory Debugging Options Tips

Summary | Leak Detection | **Heap Status** | Memory Usage | Corrupted Memory | Memory Comparisons

August 7, 2013

**Save Data**  
Export Memory Data...

**Heap Status Reports**  
Source Report  
Backtrace Report

**Other Reports Categories**  
Leak Detection Reports  
Memory Usage Reports  
Corrupted Memory Report  
Compare Memory Usage

**Other Tasks**  
Manage Filters

**Process Selection**

Process ▾

Parallel Job \_mpirun\_<mpi\_mem>.0

▣ **MPI\_COMM\_WORLD**

▣ \_mpirun\_<mpi\_mem>.0

▣ \_mpirun\_<mpi\_mem>.1

▣ \_mpirun\_<mpi\_mem>.2

**Heap Status Graphical Report**

Options

☒ Detect Leaks ☒ Relative to Baseline ☒ Enable Filtering

**Memory block:**

Type Allocated  
Filtered No  
Size 512.00KB  
Start Address 0x006cc000  
End Address 0x0074bfff  
Backtrace ID 176  
Allocator C  
Owner C

**Point of allocation:**

File ibdev\_multirail.c  
Method MPI\_SGI\_ib\_init\_slave  
Line 4677

**Guard Blocks:**

None

**Heap Information** | Backtrace/Source | Memory Content

**Overall Totals**

Category	Bytes
Heap	
▣ Corrupted Guard Blocks	0
▣ Allocated	29.00MB
▣ Deallocated	3.10KB
▣ Guard Blocks	1088
▣ Hoarded	0
▣ Leaked	Unknown

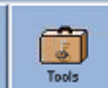
**Selected Block**

Property	Value
Start Address	0x006cc000
End Address	0x0074bfff
Size	512.00KB
Type	Allocated
Filtered	No
Backtrace ID	176
Allocator	C

**Action Points** | Processes | Threads

1 cpi.c#56 main+0x174





MemoryScope 3.4.0-0

File Tools Window Help

Home **Memory Reports** Manage Processes Memory Debugging Options TipsSummary Leak Detection **Heap Status** Memory Usage Corrupted Memory Memory Comparisons

August 7, 2013

## Save Data

Export Memory Data...

## Heap Status Reports

Source Report

Backtrace Report

## Other Reports Categories

Leak Detection Reports

Memory Usage Reports

Corrupted Memory Report

Compare Memory Usage

## Other Tasks

Manage Filters

Process Selection

Process Selection

Process **Parallel Job \_mpirun\_<mpi\_mem>.0**

- MPI\_COMM\_WORLD**
  - \_mpirun\_<mpi\_mem>.0
  - \_mpirun\_<mpi\_mem>.1
  - \_mpirun\_<mpi\_mem>.2

## Heap Status Graphical Report

Options

☒ Detect Leaks
 ☒ Relative to Baseline
 ☒ Enable Filtering


Leaked Block



Process 2: \_mpirun\_&lt;mpi\_mem&gt;.0

0x006030b0 - 0x02104c30 (27.01MB)



Heap Information

Backtrace/Source

Memory Content

Overall Totals

Category	Bytes
Heap	
<input checked="" type="checkbox"/> Corrupted Guard Blocks	0
<input checked="" type="checkbox"/> Allocated	29.00MB
<input checked="" type="checkbox"/> Deallocated	3.10KB
<input checked="" type="checkbox"/> Guard Blocks	1088
<input checked="" type="checkbox"/> Hoarded	0
<input checked="" type="checkbox"/> Leaked	Unknown

Selected Block

Property	Value
Start Address	0x006cc000
End Address	0x0074bfff
Size	512.00KB
Type	Allocated
Filtered	No
Backtrace ID	176
Allocator	C

Related Blocks

Category	Bytes
Backtrace ID 176	
<input checked="" type="checkbox"/> Allocated	512.00KB
<input checked="" type="checkbox"/> Corrupted Guard Blocks	0
<input checked="" type="checkbox"/> Deallocated	0
<input checked="" type="checkbox"/> Guard Blocks	0
<input checked="" type="checkbox"/> Hoarded	0
<input checked="" type="checkbox"/> Leaked	Unknown

Action Points Processes Threads

P-

P+

T-

T+

1 cpi.c#56 main=0x174







MemoryScope 3.4.0-0

File Tools Window Help

Home Memory Reports Manage Processes Memory Debugging Options Tips

Summary Leak Detection Heap Status

August 7, 2013

Save Data  
Export Memory Data...

Heap Status Reports  
Source Report  
Backtrace Report

Other Reports Categories  
Leak Detection Reports  
Memory Usage Reports  
Corrupted Memory Report  
Compare Memory Usage

Other Tasks  
Manage Filters

Process Selection

Process  
Parallel Job \_mpirun\_<mpi\_mem>.0  
MPI\_COMM\_WORLD  
\_mpirun\_<mpi\_mem>.0  
\_mpirun\_<mpi\_mem>.1  
\_mpirun\_<mpi\_mem>.2

Process

Property Operator Value Persistence

1 Process/Library Name contains Permanent

Exclude data matching

- any of the following
- all of the following

Evaluate

- allocation focus entry only
- all backtrace entries

Add

Remove

The conditions defined here are evaluated in the order shown, using the settings above.  
To improve performance, place the condition that will remove the most entries at the top of the list.

Persistence

- Permanent : Condition is saved for future use.
- Temporary : Condition exists for this session only, i.e. uses transient data.

Help OK Cancel

Related Blocks

Category	Bytes
Backtrace ID 176	
Allocated	512.00KB
Corrupted Guard Blocks	0
Deallocated	0
Guard Blocks	0
Hoarded	0
Leaked	Unknown

Backtrace ID 176

Allocator

Action Points Processes Threads

1 cpi.c#56 main=0x174



MemoryScope 3.4.0-0

File Tools Window Help

Home Memory Reports Manage Processes Memory Debugging Options Tips

Summary Leak Detection Heap Status

August 7, 2013

Save Data  
Export Memory Data...

Heap Status Reports  
Source Report  
Backtrace Report

Other Reports Categories  
Leak Detection Reports  
Memory Usage Reports  
Corrupted Memory Report  
Compare Memory Usage

Other Tasks  
Manage Filters

Process Selection

Process ▾  
Parallel Job \_mpirun\_<cpi\_mem>.0  
MPI\_COMM\_WORLD  
\_mpirun\_<cpi\_mem>.0  
\_mpirun\_<cpi\_mem>.1  
\_mpirun\_<cpi\_mem>.2

Filter name: MPT

Exclude data matching

- any of the following
- all of the following

Evaluate

- allocation focus entry only
- all backtrace entries

Add Remove

Property	Operator	Value	Persistence
1 Process/Library Name	contains	MPT	Permanent

The conditions defined here are evaluated in the order shown, using the settings above. To improve performance, place the condition that will remove the most entries at the top of the list.

Persistence

Permanent : Condition is saved for future use.  
Temporary : Condition exists for this session only, i.e. uses transient data.

Help OK Cancel

Related Blocks

Category	Bytes
Backtrace ID 176	
Allocated	512.00KB
Corrupted Guard Blocks	0
Deallocated	0
Guard Blocks	0
Hoarded	0
Leaked	Unknown

Action Points Processes Threads

1 cpi.c#56 main+0x174



MemoryScope 3.4.0-0

File Tools Window Help

Home Memory Reports Manage Processes Memory Debugging Options Tips

Summary Leak Detection Heap Status

August 7, 2013

Save Data  
Export Memory Data...

Heap Status Reports  
Source Report  
Backtrace Report

Other Reports Categories  
Leak Detection Reports  
Memory Usage Reports  
Corrupted Memory Report  
Compare Memory Usage

Other Tasks  
Manage Filters

Process Selection

Process  
Parallel Job \_mpirun\_<mpi\_mem>.0  
MPI\_COMM\_WORLD  
\_mpirun\_<mpi\_mem>.0  
\_mpirun\_<mpi\_mem>.1  
\_mpirun\_<mpi\_mem>.2

Opt

Proc

0x00

Hea

Over

Hea

Hoarded 0  
Leaked Unknown

Backtrace ID 176  
Allocator C

Filter name: MPT

Exclude data matching

- any of the following
- all of the following

Evaluate

- allocation focus entry only
- all backtrace entries

Add

Remove

	Property	Operator	Value	Persistence
1	Process/Library Name	contains	MPT	Permanent
2	Process/Library Name	contains		Permanent

Add another condition to the filter.

Help OK Cancel

Leaked Block

Related Blocks

Category	Bytes
Backtrace ID 176	
Allocated	512.00KB
Corrupted Guard Blocks	0
Deallocated	0
Guard Blocks	0
Hoarded	0
Leaked	Unknown

Action Points Processes Threads

1 cpi.c#56 main+0x174



MemoryScope 3.4.0-0

File Tools Window Help

Home Memory Reports Manage Processes Memory Debugging Options Tips

Summary Leak Detection Heap Status

August 7, 2013

Save Data  
Export Memory Data...

Heap Status Reports  
Source Report  
Backtrace Report

Other Reports Categories  
Leak Detection Reports  
Memory Usage Reports  
Corrupted Memory Report  
Compare Memory Usage

Other Tasks  
Manage Filters

Process Selection

Process ▾  
Parallel Job \_mpirun\_<cpi\_mem>.0  
MPI\_COMM\_WORLD  
\_mpirun\_<cpi\_mem>.0  
\_mpirun\_<cpi\_mem>.1  
\_mpirun\_<cpi\_mem>.2

Filter name: MPT

Exclude data matching  
any of the following  
all of the following

Evaluate  
allocation focus entry only  
all backtrace entries

Add  
Remove

	Property	Operator	Value	Persistence
1	Process/Library Name	contains	MPT	Permanent
2	Process/Library Name	contains	lib	Permanent

The conditions defined here are evaluated in the order shown, using the settings above. To improve performance, place the condition that will remove the most entries at the top of the list.

Persistence  
Permanent : Condition is saved for future use.  
Temporary : Condition exists for this session only, i.e. uses transient data.

Help OK Cancel

Leaked Block

Related Blocks

Category	Bytes
Backtrace ID 176	
Allocated	512.00KB
Corrupted Guard Blocks	0
Deallocated	0
Guard Blocks	0
Hoarded	0
Leaked	Unknown

Action Points Processes Threads

1 cpi.c#56 main+0x174



The screenshot displays a Linux desktop environment accessed via TightVNC. The desktop features a taskbar at the top with icons for file manager, a clock, and several application windows. A terminal window in the background shows a list of processes with columns for PID, PPID, user, and process name. The foreground is dominated by the MemoryScope 3.4.0-0 application, which is used for memory debugging. The main window of MemoryScope shows a 'Heap Status' report for a process named 'Parallel Job \_mpirun\_<mpi.mpt6>.0'. A 'Memory Debugging Data Filters' dialog box is open, showing a table of filters. The table has columns for 'Filter Name', 'Owner', and 'Data Filtered'. One filter is listed: 'MPT' with owner 'User' and data filtered '0'. The dialog also includes buttons for 'Add...', 'Edit...', 'Remove', 'Enable All', 'Disable All', 'Ok', and 'Cancel'. In the background, a file manager window shows a directory structure with files like '2028809.pbsp11', '2028810.pbsp11', etc. A terminal window shows the output of a command, including '2,so'...done' and 'initially loaded at 0xff0'.

Process Selection

Process
Parallel Job _mpirun_<mpi.mpt6>.0
MPComm_WORLD
_mpirun_<mpi.mpt6>.0
_mpirun_<mpi.mpt6>.1
_mpirun_<mpi.mpt6>.2

MemoryScope 3.4.0-0

File Tools Window Help

Home Memory Reports Manage Processes Memory Debugging Options Tips

Summary | Leak Detection | **Heap Status** | Memory Usage | Corrupted Memory | Memory Comparisons

August 16, 2013

Save Data  
Export Memory Data...

Heap Status Reports  
Source Report  
Backtrace Report

Other Reports Categories  
Leak Detection Reports  
Memory Usage Reports  
Corrupted Memory Report  
Compare Memory Usage

Other Tasks  
Manage Filters

Memory Debugging Data Filters

Enabled filters are applied in the order shown:

Filter Name	Owner	Data Filtered
<input checked="" type="checkbox"/> MPT	User	0

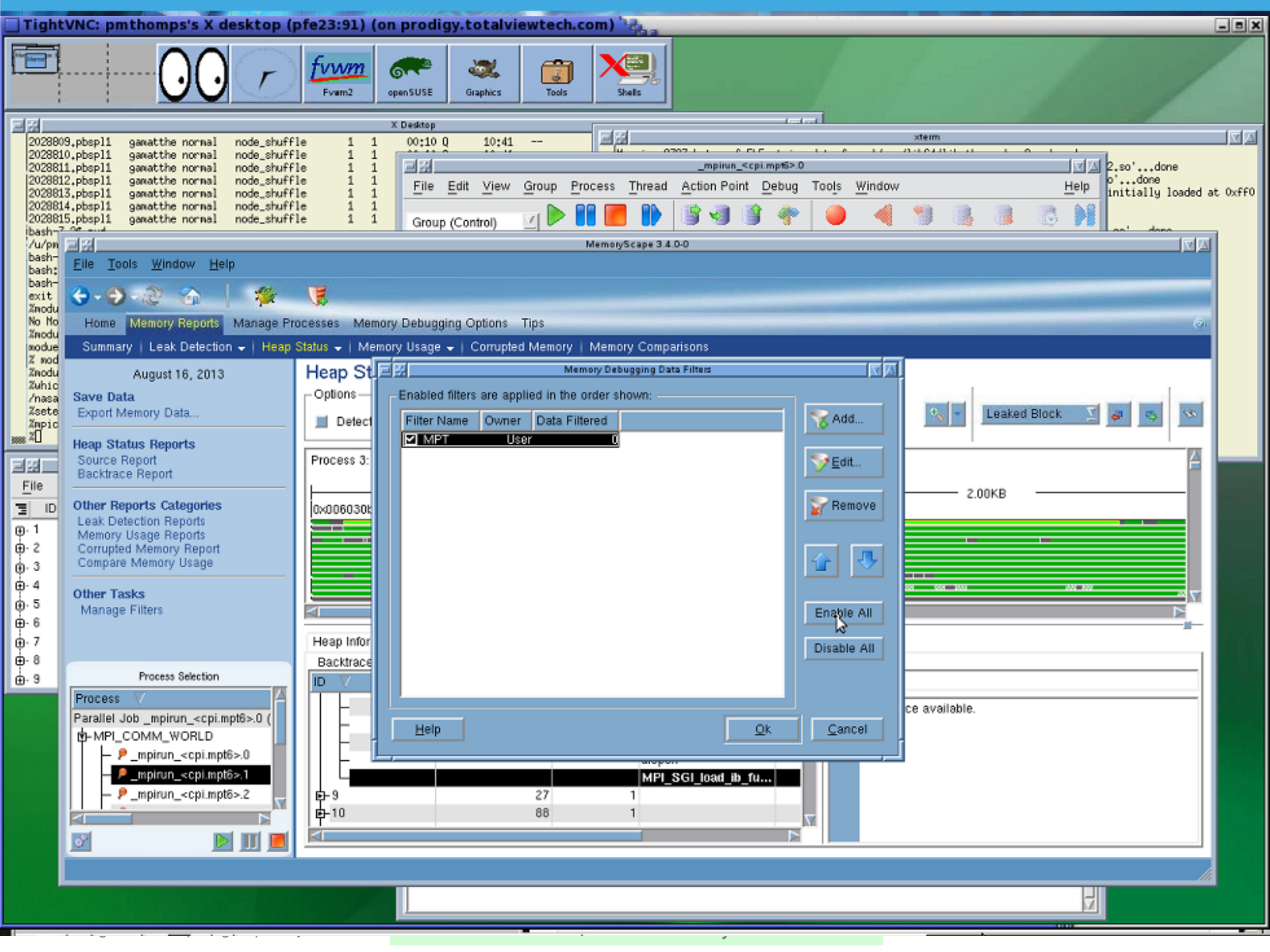
Buttons: Add..., Edit..., Remove, Enable All, Disable All, Ok, Cancel

Process 3: 0x006030K

Heap Information

ID	Backtrace
9	27 1 MPI_SGI_load_ib_fu...
10	88 1







The screenshot displays a Linux desktop environment accessed via TightVNC. The desktop features a taskbar with icons for 'Fvwm2', 'openSUSE', 'Graphics', 'Tools', and 'Shells'. A terminal window in the background shows a list of processes, including 'node\_shuffle' and 'node\_shuffle'. A file manager window displays a directory structure. The foreground application is 'MemoryScope 3.4.0-0', which is used for memory debugging. It shows a 'Heap Status' report for 'Process 3: 0x006030k'. A 'Memory Debugging Data Filters' dialog box is open, showing a table of filters:

Filter Name	Owner	Data Filtered
<input checked="" type="checkbox"/> MPT	User	0

The dialog box also includes buttons for 'Add...', 'Edit...', 'Remove', 'Enable All', 'Disable All', 'Help', and 'Cancel'. The main window of MemoryScope shows a 'Leaked Block' of 2.00KB and a memory usage graph. The bottom of the screen shows a 'Process Selection' dialog box with a tree view of processes, including 'Parallel Job \_mpirun\_<mpi.mpt6>.0' and its sub-processes.



**Process Selection**

- Process ▾
- Parallel Job \_mpirun\_<cpu.mpt6>.0
- ▾ MPI\_COMM\_WORLD
  - \_mpirun\_<cpu.mpt6>.0
  - \_mpirun\_<cpu.mpt6>.1**
  - \_mpirun\_<cpu.mpt6>.2

**Heap Status Graphical Report**

Options:

- ☐ Detect Leaks
- ☐ Relative to Baseline
- ☒ Enable Filtering

Process 3: \_mpirun\_<cpu.mpt6>.1

0x006030b0 - 0x01328cd0 (13.15MB)

2.00KB

**Heap Information** | Backtrace/Source | Memory Content

Backtrace

ID ▾	Total Bytes	Count	Function	Line
6	5.40KB	36		
7	27	1		
8	1176	1		
9	27	1		
10	88	1		
11	264	1		
12	168	1		
13	174	1		

Source



Memory Usage

X Desktop

2028809.pbsp11	ganatthe	normal	node_shuffle	1	1	00:10	0	10:41	--
2028810.pbsp11	ganatthe	normal	node_shuffle	1	1				
2028811.pbsp11	ganatthe	normal	node_shuffle	1	1				
2028812.pbsp11	ganatthe	normal	node_shuffle	1	1				
2028813.pbsp11	ganatthe	normal	node_shuffle	1	1				
2028814.pbsp11	ganatthe	normal	node_shuffle	1	1				
2028815.pbsp11	ganatthe	normal	node_shuffle	1	1				

xterm

2,so'...done  
o'...done  
initially loaded at 0xff0

\_mpirun\_<mpi.mpt6>.0

File Edit View Group Process Thread Action Point Debug Tools Window Help

Group (Control)

MemoryScope 3.4.0-0

File Tools Window Help

Home Memory Reports Manage Processes Memory Debugging Options Tips

Summary Leak Detection Heap Status Memory Usage Corrupted Memory Memory Comparisons

August 16, 2013

Save Data  
Export Memory Data...

Heap Status Reports  
Source Report  
Backtrace Report

Other Reports Categories  
Leak Detection Reports  
Memory Usage Reports  
Corrupted Memory Report  
Compare Memory Usage

Other Tasks  
Manage Filters

Process Selection

Parallel Job \_mpirun\_<mpi.mpt6>.0

MPI\_COMM\_WORLD

\_mpirun\_<mpi.mpt6>.0

**\_mpirun\_<mpi.mpt6>.1**

\_mpirun\_<mpi.mpt6>.2

Heap Status Graphical Report

Options

☒ Detect Leaks ☒ Relative to Baseline ☐ Enable Filtering

\_mpirun\_<mpi.mpt6>.1

2.00KB

x01328cd0 (13.15MB)

Heap Information Backtrace/Source Memory Content

Backtrace

ID	Total Bytes	Count	Function	Line
6	5.40KB	36		
7	27	1		
8	1176	1		
9	27	1		
10	88	1		
11	264	1		
12	168	1		
1.4	17.4	1.4		

Source

Memory block:

Type Allocated

Filtered No

Size 150

Start Address 0x00608c90

End Address 0x00608d25

Backtrace ID 220

Allocator C

Owner C

Point of allocation:

File cpi.c

Method main

Line 62

Guard Blocks:

Pre-guard

size 16 bytes

pattern 0x77777777

Post-guard

size 16 bytes

pattern 0x99999999



The screenshot displays a Linux desktop environment accessed via TightVNC. The top taskbar contains icons for file manager, a clock, and several application shortcuts including 'Fvwm2', 'openSUSE', 'Graphics', 'Tools', and 'Shells'. A terminal window in the background shows a list of system logs with columns for timestamp, user, session, and process. In the foreground, the 'MemoryScope 3.4.0-0' application is open, displaying a 'Heap Status Graphical Report'. The report includes a summary of memory usage (13.15MB) and a detailed backtrace table. A 'Process Selection' dialog is also visible, showing a tree of running processes.

**Terminal Window (Background):**

Timestamp	User	Session	Process
2028809.pbasp11	ganatthe	normal	node_shuffle
2028810.pbasp11	ganatthe	normal	node_shuffle
2028811.pbasp11	ganatthe	normal	node_shuffle
2028812.pbasp11	ganatthe	normal	node_shuffle
2028813.pbasp11	ganatthe	normal	node_shuffle
2028814.pbasp11	ganatthe	normal	node_shuffle
2028815.pbasp11	ganatthe	normal	node_shuffle

**MemoryScope 3.4.0-0 - Heap Status Graphical Report**

Options: ☒ Detect Leaks ☒ Relative to Baseline ☐ Enable Filtering

Summary | Leak Detection | **Heap Status** | Memory Usage | Corrupted Memory | Memory Comparisons

August 16, 2013

Save Data  
Export Memory Data...

**Heap Status Reports**  
Source Report  
Backtrace Report

**Other Reports Categories**  
Leak Detection Reports  
Memory Usage Reports  
Corrupted Memory Report  
Compare Memory Usage

**Other Tasks**  
Manage Filters

**Process Selection**

- Process
- Parallel Job \_mpirun\_<mpi.mpt6>.0
- MPI\_COMM\_WORLD
- \_mpirun\_<mpi.mpt6>.0
- \_mpirun\_<mpi.mpt6>.1**
- \_mpirun\_<mpi.mpt6>.2

**Backtrace**

ID	Total Bytes	Count	Function	Line
217	40	1		
218	2.00KB	1		
219	32	1		
220	450	3		
			malloc	
			main	
			_libc_start_main	
			_start	

**Source**

```

/u/pmthomp/cpi.c
57
58
59
60     x = h * ((double)i - 0.5);
61     sum += f(x);
62     freeptr = (char *) malloc ((unsigned long)150 * s
63     strcpy/freeptr. "Testing":
  
```







# More Information



MemoryScape demonstration videos available on the  
Rogue Wave TotalView Products page

<http://www.roguewave.com/products/totalview/resources/videos.aspx>